

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ  
СІКОРСЬКОГО»**

Теплоенергетичний факультет

Кафедра автоматизації проектування енергетичних процесів і систем

"На правах рукопису"  
УДК 004.912

«До захисту допущено»  
Завідувач кафедри

\_\_\_\_\_ О.В. Коваль  
(підпис) (ініціали, прізвище)

“ \_\_\_\_ ” \_\_\_\_\_ 2018р.

## Магістерська дисертація

зі спеціальності - 122 Комп'ютерні науки та інформаційні технології  
за спеціалізацією - Геометричне моделювання в інформаційних системах  
на тему: «Модуль внутрішніх повідомлень веб - системи управління  
педагогічними та науковими аспектами роботи кафедри»

Виконав: студент 6 курсу, групи ТР-71мп  
Лучков Владислав Феліксович  
(прізвище, ім'я, по батькові)

\_\_\_\_\_  
(підпис)

Науковий керівник  
доцент., к.т.н. Карпенко Є. Ю.  
(посада, вчене звання, науковий ступінь, прізвище та ініціали)

\_\_\_\_\_  
(підпис)

Рецензент

\_\_\_\_\_  
(посада, вчене звання, науковий ступінь, прізвище та ініціали)

\_\_\_\_\_  
(підпис)

Засвідчую, що у цій магістерській  
дисертації немає запозичень з праць  
інших авторів без відповідних  
посилань.

Студент \_\_\_\_\_  
(підпис)

Київ - 2018

**Національний технічний університет України  
“Київський політехнічний інститут ім. Ігоря Сікорського”**

Факультет теплоенергетичний

Кафедра автоматизації проектування енергетичних процесів і систем

Рівень вищої освіти другий, магістерський

зі спеціальності - 122 Комп'ютерні науки та інформаційні технології

за спеціалізацією - Геометричне моделювання в інформаційних системах

ЗАТВЕРДЖУЮ  
Завідувач кафедри  
Коваль О.В. \_\_\_\_\_  
(прізвище, ініціали) (підпис)  
«\_\_\_\_» \_\_\_\_\_ 2018р.

**З А В Д А Н Н Я  
НА МАГІСТЕРСЬКУ ДИСЕРТАЦІЮ СТУДЕНТУ**

Лучков Владислав Феліксович

(прізвище, ім'я, по батькові)

1. Тема дисертації Модуль внутрішніх повідомлень веб - системи управління педагогічними та науковими аспектами роботи кафедри

Науковий керівник Карпенко Євген Юрійович, к.т.н., доц.

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від “\_\_\_” \_\_\_\_\_ 20\_\_ року №\_\_\_

2. Строк подання студентом дисертації \_\_\_\_\_

3. Об'єкт дослідження: веб - системи управління педагогічними та науковими аспектами

4. Предмет дослідження: веб - системи управління педагогічними та науковими аспектами

5. Перелік питань, які потрібно розробити \_\_\_\_\_

– визначити процеси, які відбуваються на кафедрі навчального закладу під час навчання;

– дослідити роль користувачів та їх функції стосовно представлення науково-методичних робіт кафедри;

– спроєктувати роботу основних об'єктів модуля, представити алгоритми їх роботи;

– виконати побудову веб орієнтованого модуля, що буде надавати можливість управління педагогічними та науковими роботами кафедри з можливістю формування повідомлень

6. Орієнтований перелік ілюстративного матеріалу презентація на тему «Модуль внутрішніх повідомлень веб системи управління науковими та педагогічними аспектами роботи кафедри»

7. Орієнтований перелік публікацій \_\_\_\_\_

1. «Засоби корегування планів навчання до потреб ІТ ринку праці» // Сучасні проблеми наукового забезпечення енергетики: Матеріали XVI Міжнародної науково-практичної конференції аспірантів, магістрантів і студентів, м. Київ, 24–27 квітня 2018 р. У 2 т. – К. : 7 КПП ім. Ігоря Сікорського», 2018. – Т. 2. – 194с.

8. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Основна частина	Карпенко Є.Ю., доц., к.т.н.		

9. Дата видачі завдання «\_\_\_\_\_»\_\_\_\_\_2018 р.

### КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання магістерської дисертації	Строки виконання етапів магістерської дисертації	Примітка
1	Отримання завдання	19.01.18 р.	
2	Аналіз вимог завдання, вибір методів і засобів розв’язання поставленої задачі	20.01.18р. — 28.01.18р.	
3	Підготовка матеріалів магістерської роботи	12.10.18р. — 21.11.18р.	
4	Підготовка публікацій	03.04.18р. — 06.04.18р.	
5	Доповідь на конференції	27.04.18р.	
6	Написання основних розділів автореферату	6.11.18р. — 20.11.18р.	
7	Захист програмного продукту	25.10.18р	
8	Передзахист	26.11.18р	
9	Захист	17.12.18р	

Студент

\_\_\_\_\_  
( підпис )

\_\_\_\_\_  
Лучков В.Ф.

\_\_\_\_\_  
(прізвище та ініціали)

Науковий керівник

\_\_\_\_\_  
( підпис )

\_\_\_\_\_  
Карпенко Є.Ю.

\_\_\_\_\_  
(прізвище та ініціали)

## РЕФЕРАТ

**Структура роботи.** Магістерська дисертація складається зі вступу, трьох розділів, висновку, переліку посилань з 50 найменувань, 4 додатки, і містить 42 рисунків, 42 таблиці. Повний обсяг магістерської дисертації складає 112 сторінок, з яких перелік посилань займає 4 сторінки, додатки – 6 сторінок.

**Актуальність теми.** Мережа Інтернет надає велику кількість інформації, необхідну для навчання, студент може знайти лекції, практичні завдання, методичні вказівки, виконані курсові роботи. Але при цьому порушуються авторські права, поширюється несанкціоноване розповсюдження. Тому розробка веб системи яка надає науковий та педагогічний матеріал в рамках тільки спеціальності, факультету або вишу дозволить вирішити дві задачі – надати необхідний для навчання матеріал та запобігти використування цих матеріалів студентами інших вишів.

**Метою дослідження** є проектування та програмна розробка модуля внутрішніх повідомлень веб - системи управління педагогічними та науковими аспектами роботи кафедри.

Для досягнення поставленої мети були сформульовані наступні **завдання дослідження**, що визначили логіку дослідження та його структуру:

- визначити процеси, які відбуваються на кафедрі навчального закладу під час навчання;
- визначити які матеріали буде розміщено в модулі, що розробляється;
- дослідити роль користувачів та їх функції стосовно представлення науково-методичних робіт кафедри;
- сформувати вимоги до розробки модуля внутрішніх повідомлень веб - системи управління педагогічними та науковими аспектами роботи кафедри;
- визначити платформу розробки модуля;
- спроектувати роботу основних об'єктів модуля, представити алгоритми їх роботи;
- виконати побудову веб орієнтованого модуля, що буде надавати можливість управління педагогічними та науковими роботами кафедри з можливістю формування повідомлень;

- протестувати отримані результати.

**Об'єктом дослідження** є принципи розробки веб орієнтованих додатків представлення науково-методичного матеріалу в мережі Інтернет з розподіленими правами доступу до ресурсів.

**Предметом дослідження** є принципи розробки веб системи управління педагогічними та науковими роботами кафедри.

**Методи дослідження.** Розв'язання поставлених задач виконувались з використанням наступних методів:

- аналізу процесів в нотації IDEF0;
- аналізу процесів в нотації DFD;
- побудові моделі за допомогою діаграм мови UML;
- розробка програмного коду мовою PHP та бази даних за допомогою СУБД MySQL.

**Наукова новизна одержаних результатів.** Найбільш суттєвими науковими результатами магістерської дисертації є:

- побудова моделей формування бізнес процесів, виникаючих при роботі кафедри;
- розроблено моделі алгоритмічного забезпечення веб системи управління науковими та педагогічними роботами кафедри.

**Практичне значення одержаних результатів** роботи полягає в підвищенні ефективності навчання за рахунок надання необхідних науково-методичних матеріалів, модуль повідомлень дозволить підтримувати зворотній зв'язок під час навчання, підвищить рівень комунікації між викладачем та студентами.

**Ключові слова.** *МОДЕЛЮВАННЯ, ВЕБ СИСТЕМА, КАФЕДРА, ПЕДАГОГІЧНИЙ МАТЕРІАЛ, МОВА PHP, ВИКЛАДАЧ, СТУДЕНТ, НАУКОВА РОБОТА, СХОВИЩЕ ДАНИХ.*

# ABSTRACT

## on master's thesis

**Structure.** The master's dissertation consists of an introduction, three sections, a conclusion, a list of references from 50 titles, 4 annexes, and contains 42 figures, 42 tables. The full volume of the master's thesis is 112 pages, of which the list of links takes 4 pages, applications - 6 pages.

**Actuality of theme.** The Internet network provides a large amount of information necessary for training, the student can find lectures, practical tasks, methodical instructions, coursework. But this violates copyright, spreads unauthorized distribution. Therefore, the development of a web system that provides scientific and pedagogical material within the framework of a specialty, faculty or higher will solve two problems - to provide the necessary material for study and to prevent the use of these materials by students from other universities.

**The purpose of the research** is to design and program development of the module of internal messages of the web - system of teaching pedagogical and scientific aspects of the department work.

**To achieve this goal**, the following research objectives were formulated, which determined the logic of the research and its structure:

- to determine the processes that take place at the department of the educational institution during the training;
- determine which materials will be placed in the module being developed;
- to investigate the role of users and their functions in relation to the presentation of scientific and methodological works of the department;
- to formulate requirements for the development of the module of internal messages of the web - system of teaching pedagogical and scientific aspects of the department work;
- define the platform for the development of the module;
- to design the work of the main objects of the module, to present the algorithms of their work;

- to carry out the construction of a web-based module, which will provide the opportunity to manage the pedagogical and scientific work of the department with the possibility of forming messages;

- test the results.

**The object of the study** is the principles of developing web-based applications for the presentation of scientific and methodological material on the Internet with distributed rights of access to resources.

**The subject of the study** is the principles of developing a web-based system for teaching pedagogical and scientific work of the department.

**Research methods.** The solution of the set tasks was carried out using the following methods:

- analysis of processes in the notation IDEF0;
- analysis of processes in the DFD notation;
- construction of the model using UML diagrams;
- Software code development in PHP and database using MySQL database.

**Scientific novelty of the obtained results.** The most significant scientific results of the master's thesis are:

- construction of models of formation of business processes that arise at the work of the department;
- models of algorithmic maintenance of the web system of management of scientific and pedagogical works of the department have been developed.

**The practical significance of the results** of the work is to increase the effectiveness of the training by providing the necessary scientific and methodological materials, the module of messages will allow to maintain feedback during the training, will increase the level of communication between the teacher and the students.

**Keywords.** *MODELING, WEB SYSTEM, PEDAGOGICAL MATERIAL, PHP LANGUAGE, TEACHER, STUDENT, SCIENTIFIC WORK, DATA SHEET.*

## ЗМІСТ

Вступ.....	11
1. Аналіз наукових та педагогічних аспектів роботи кафедри .....	14
1.1. Опис наукових та педагогічних робіт кафедри .....	14
1.2. Аналіз процесів під час функціонування кафедри навчального закладу .....	16
1.3. Постановка задачі на розробку модуля внутрішніх повідомлень веб системи управління науковими та педагогічними аспектами роботи кафедри.....	27
1.4. Визначення етапів розробки веб системи управління науковими та педагогічними аспектами роботи кафедри .....	30
Висновки по розділу 1 .....	37
2. Розробка проекту веб системи управління науковими та педагогічними аспектами роботи кафедри з модулем формування повідомлень .....	39
2.1. Опис функції користувачів системи .....	39
2.2. Розробка алгоритмічного забезпечення веб системи управління.....	45
2.3. Вибір платформи .....	54
Висновки по розділу 2 .....	66
3. Опис програмної розробки веб системи управління науковою та педагогічною роботою кафедри з модулем формування повідомлень.....	68
3.1. Побудова діаграми класів .....	68
3.2. Побудова діаграми виконання.....	72
3.3. Побудова логічної моделі бази даних веб системи.....	74
3.4. Вибір середовища розробки.....	79
Висновки по розділу 3 .....	82
4. Методика роботи користувача з програмною системою .....	83
Висновок по розділу 4 .....	88
5. Розроблення стартап-проекту .....	89
5.1. Опис ідеї стартап-проекту .....	89
5.2. Технологічний аудит ідеї проекту.....	91
5.3. Аналіз ринкових можливостей запуску стартап-проекту .....	92



5.4. Розроблення ринкової стратегії проекту .....	101
5.5. Розроблення маркетингової програми стартап-проекту .....	104
Висновки до розділу 5 .....	106
Висновки.....	108
Список використаних джерел.....	109
Додаток А .....	<b>Ошибка! Закладка не определена.</b>
Додаток Б.....	<b>Ошибка! Закладка не определена.</b>

## ВСТУП

Забезпечення комунікаціями є необхідним процесом для будь-якої галузі життя, своєчасно отримана достовірна інформація дозволяє приймати вірні обґрунтовані оперативні рішення.

Особливо актуальними комунікації стають під час навчання в навчальних закладах – отримана інформація від викладача є запорукою вдалого навчання студента, інформація від студента також необхідна викладачу для вдосконалення навчального процесу, надання необхідних знань та вмінь.

На даний момент більшість комунікацій перенесено у мережу Інтернет – різноманітні соціальні мережі, форуми, блоги, електронна пошта, все це базується на використанні веб додатків.

Актуальність роботи. Мережа Інтернет надає велику кількість інформації, необхідну для навчання, студент може знайти лекції, практичні завдання, методичні вказівки, виконані курсові роботи. Але при цьому порушуються авторські права, поширюється несанкціоноване розповсюдження. Тому розробка веб системи, яка надає науковий та педагогічний матеріал в рамках тільки спеціальності, факультету або вишу дозволить вирішити дві задачі – надати необхідний для навчання матеріал та запобігти використування цих матеріалів студентами інших вишів.

Предметом дослідження є принципи розробки веб системи управління педагогічними та науковими роботами кафедри.

Об'єкт дослідження принципи розробки веб орієнтованих додатків представлення науково-методичного матеріалу в мережі Інтернет з розподіленими правами доступу до ресурсів.

Розробка даної системи дасть можливість забезпечити навчальними, методичними та науковими матеріалам студентів навчального закладу.

Метою роботи є розробка модуля внутрішніх повідомлень веб - системи управління педагогічними та науковими аспектами роботи кафедри.

Для досягнення поставленої мети в рамках цієї роботи необхідно вирішити

наступні задачі:

- визначити процеси, які відбуваються на кафедрі навчального закладу під час навчання;
- визначити які матеріали буде розміщено в модулі, що розробляється;
- дослідити роль користувачів та їх функції стосовно представлення науково-методичних робіт кафедри;
- сформулювати вимоги до розробки модуля внутрішніх повідомлень веб - системи управління педагогічними та науковими аспектами роботи кафедри;
- визначити платформу розробки модуля;
- спроектувати роботу основних об'єктів модуля, представити алгоритми їх роботи;
- виконати побудову веб орієнтованого модуля, що буде надавати можливість управління педагогічними та науковими роботами кафедри з можливістю формування повідомлень;
- протестувати отримані результати.

Теоретична значимість роботи – розроблені моделі можуть бути застосовані для представлення різноманітного теоретичного та практичного матеріалу, та застосовані при побудові систем зберігання та формування повідомлень між учасниками навчання.

Практична значимість – підвищення ефективності навчання за рахунок надання необхідних науково-методичних матеріалів, за рахунок використання веб інструментів, модуль повідомлень дозволить підтримувати зворотній зв'язок під час навчання, підвищить рівень комунікації між викладачем та студентами.

Робота представлена вступом, основною частиною, висновками, літературою, додатками.

Основна частина представлена п'ятьма розділами.

У першому розділі представлений аналіз процесів функціонування кафедри, аналіз представлено за допомогою діаграм IDEF0, DFD. Визначені вимоги до побудови веб системи управління педагогічними та науковими аспектами роботи

кафедри. Визначено етапи та стадії розробки веб системи відповідно до стандартів розробки веб додатків та інформаційних систем.

У другому розділі представлено опис функції користувачів, що будуть працювати з системою, для цього були застосовані діаграми варіантів використання, представлено алгоритмічне забезпечення системи. Проведено аналіз існуючих платформ розробки веб систем, проведено обґрунтований аналіз, який базується на методі аналізу ієрархії.

У третьому розділі представлена діаграму класів, компонентів, розгортання. На основі аналізу в другий частини визначено, що найкращим варіантом буде розробка веб системи шляхом написання власного коду, для цього проведено аналіз можливостей мови сценаріїв PHP та СУБД MySQL. Сформовано карту сайту.

У четвертому розділі представлено опис функціонування розробленої веб системи управління науковими та педагогічними аспектами роботи кафедри з модулем повідомлень.

У п'ятому розділі представлено описання розроблення стартап проекту.

# **1 АНАЛІЗ НАУКОВИХ ТА ПЕДАГОГІЧНИХ АСПЕКТІВ РОБОТИ КАФЕДРИ**

## **1.1 Опис наукових та педагогічних робіт кафедри**

Науково-педагогічні роботи – роботи, які направлені на підвищення наукового, професійного та практичного рівня знань як у студентів, так і безпосередньо у викладачів.

Наукові та педагогічні роботи кафедри можна умовно поділити наступним чином:

- роботи внутрішнього використання – роботи, які можуть бути доступні тільки викладачам кафедри, керівництву факультету, вишу або інших кафедр (за згодою);

- роботи для загального користування – роботи, які в першу чергу спрямовані на навчальний процес та доступні студентам.

Визначимо які роботи можуть бути надані для загального користування:

- конспект лекцій;
- методичні вказівки по практичним роботам;
- методичні вказівки по лабораторним роботам;
- програма та рекомендації по навчальній та виробничій практиці»
- методичні вказівки по курсовим роботам;
- методичні вказівки по розрахунково графічним роботам;
- методичні посібники по переддипломній практиці;
- рекомендації щодо виконання бакалаврських та магістерських робіт;
- завдання до самостійної роботи;
- питання до іспиту або заліку;
- розклад занять;
- накази на практику, до дипломування та інші;
- перелік рекомендованої літератури по дисципліні;

- правила оформлення робіт, норми ДСТУ;
- правила формування наукових статей, тез, доповідей;
- інформація про наукові студентські конференції;
- збірники наукових робіт, які мають вільне розповсюдження.

До робіт внутрішнього використання належать:

- робочі програми;
- екзаменаційні білети;
- індивідуальні плани;
- навчальні програми;
- звіти про навантаження;
- протоколи засідань кафедри;
- звіти виконаних робіт;
- завдання до ККР;
- монографії, підручники та інші наукові праці;
- конференції, в яких приймає участь кафедра;
- перелік наукових публікацій;
- навантаження викладачів;
- розклад занять;
- внутрішні розпорядження по вишу;
- посадові інструкції;
- наукові збірники;
- автореферати викладачів, що мають науковий ступень;
- внутрішня інформація кафедри – відрядження, відпустки;
- та інше.

Як видно із переліку для організації наукової та педагогічної роботи кафедри необхідно переробляти велику кількість даних, для запобігання її втрати необхідно застосовувати надійне сховище, при цьому інформація може змінюватися достатньо часто – наприклад методичні вказівки до лабораторних робіт необхідно хоча раз в два роки, а ось екзаменаційні квітки щорічно [7].

## **1.2 Аналіз процесів підчас функціонування кафедри навчального закладу**

Визначемо основні процеси, для наочності відображення процесів скористаємося діаграмами опису IDEF. Стандарт IDEF використовується як інструмент для створення комп'ютерних програм, що підтримують виробничі процеси, а також як інструмент, що дозволяє складати діаграми процесів в сервісних і виробничих компаніях [29]. Стандарт IDEF дозволяє визначити масштаб аналізу і побудувати відносини між аналітиком і його клієнтом. IDEF використовується як засіб комунікації через нормалізованих графічних символів і як інструмент аналізу, показуючи, які дії повинні виконуватися для правильного моделювання процесу. Стандарт IDEF0 також допомагає визначити сильні і слабкі сторони, що моделюється [30].

Побудова процесу за допомогою цих діаграм починається з розробки контекстної діаграми, яка представляє собою один блок, що має наступні інформаційні потоки [32]:

- вхідні потоки – стрілки ліворуч;
- вихідні потоки – стрілки праворуч;
- управління – стрілки зверху;
- механізми – стрілки знизу.

На рисунку 1.1 представлена контекстна діаграма «Ведення наукової та педагогічної роботи кафедри».

На рисунку представлені наступні інформаційні потоки:

### **1. Вхідні дані:**

- співробітники кафедри;
- студенти, що навчаються у викладачів кафедри;
- наукові праці;
- навчальні матеріали.

### **2. Вихідні дані:**

- перелік наукових праць;

- навчально-методична документація;
- звіти.

Механізми – MS Office.

Управління – законодавство, внутрішні акти, інструкції.

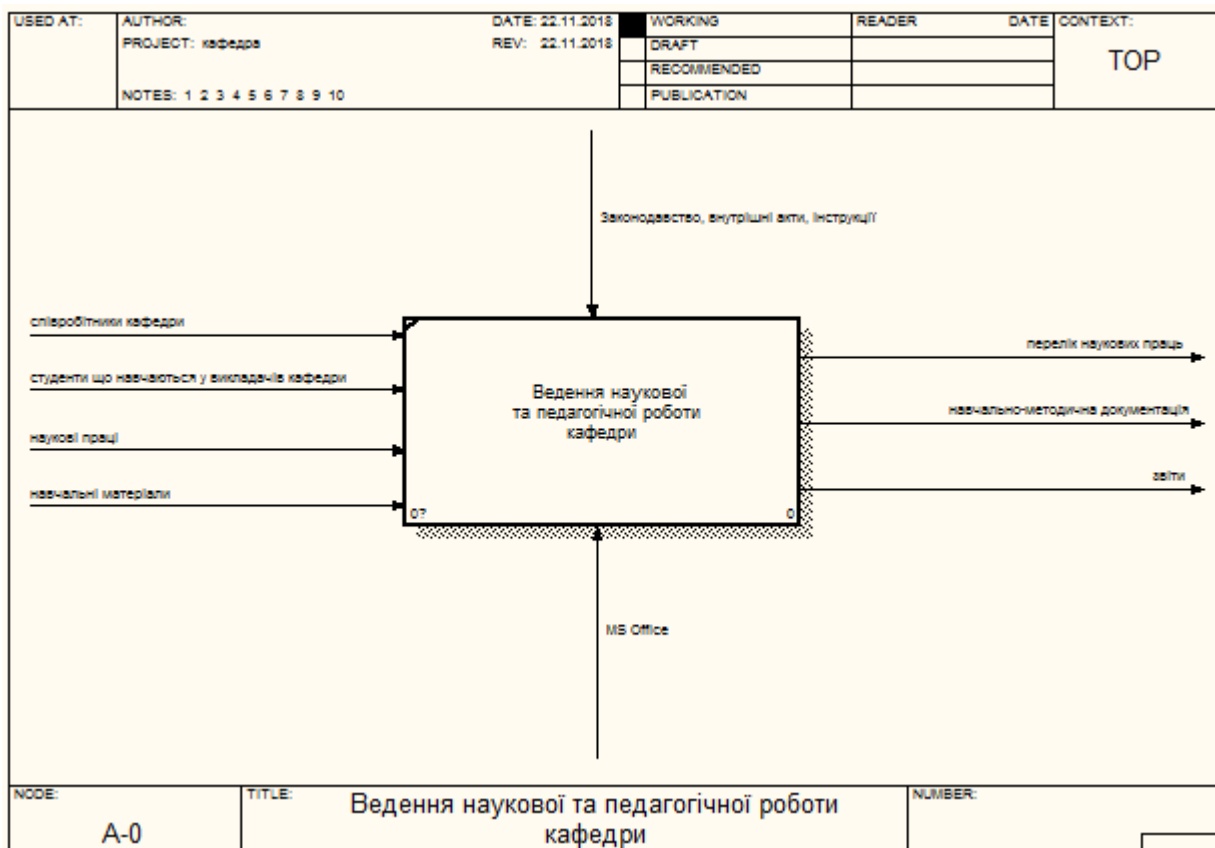


Рисунок 1.1 – Контекстна діаграма «Введення наукової та педагогічної роботи кафедри»

В подальшому процес деталізується – розбивається на більш детальні блоки для більш глибокого представлення процесу [29]. На рисунку 1.2 представлена деталізація контекстної діаграми, вона складається з наступних блоків:

- обробка інформації по співробітникам;
- формування інформації про студентів кафедри;
- обробка наукових робіт кафедри;
- обробка навчально-методичних матеріалів.



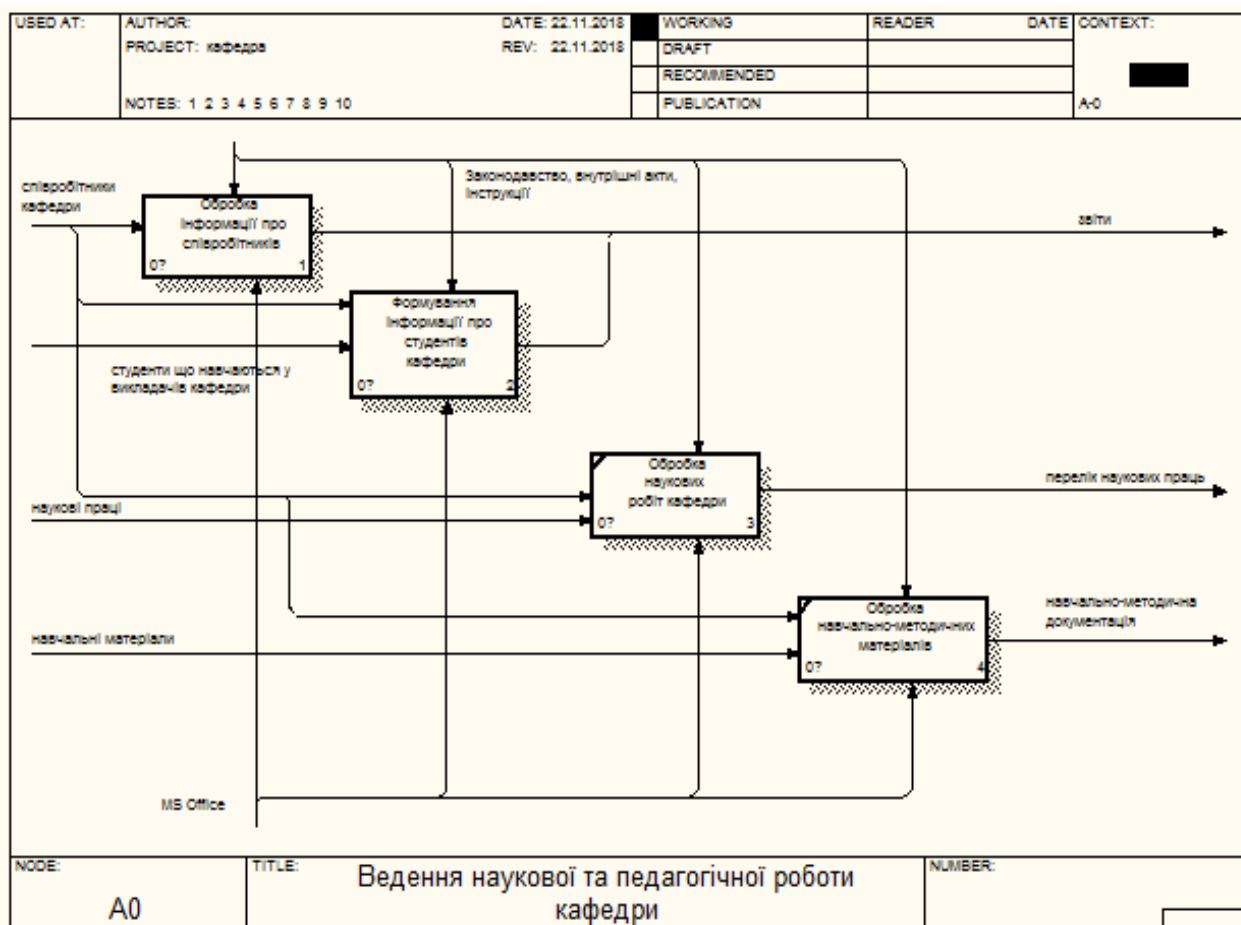


Рисунок 1.2 Декомпозиція контекстної діаграми

На рисунку 1.3 представлено декомпозицію процесу «Обробка інформації про співробітників». Цей процес представлено наступними блоками:

- формування індивідуальних даних по викладачам кафедри, цей блок має вхід співробітники кафедри, вихід дані, управління законодавство, внутрішні акти, інструкції, механізмом є MS Office;
- формування документації для факультету або вишу, цей блок має такі самі інформаційні потоки: вхід співробітники кафедри, вихід дані, управління законодавство, внутрішні акти, інструкції, механізмом є MS Office;
- зберегти дані по співробітникам, цей блок має вхід дані, вихід інформація для друку, управління законодавство, внутрішні акти, інструкції, механізмом є MS Office;

– роздрукувати для перевірки або користування, цей блок має вхід інформація для друку, вихід звіти, управління законодавство, внутрішні акти, інструкції, механізмом є MS Office.

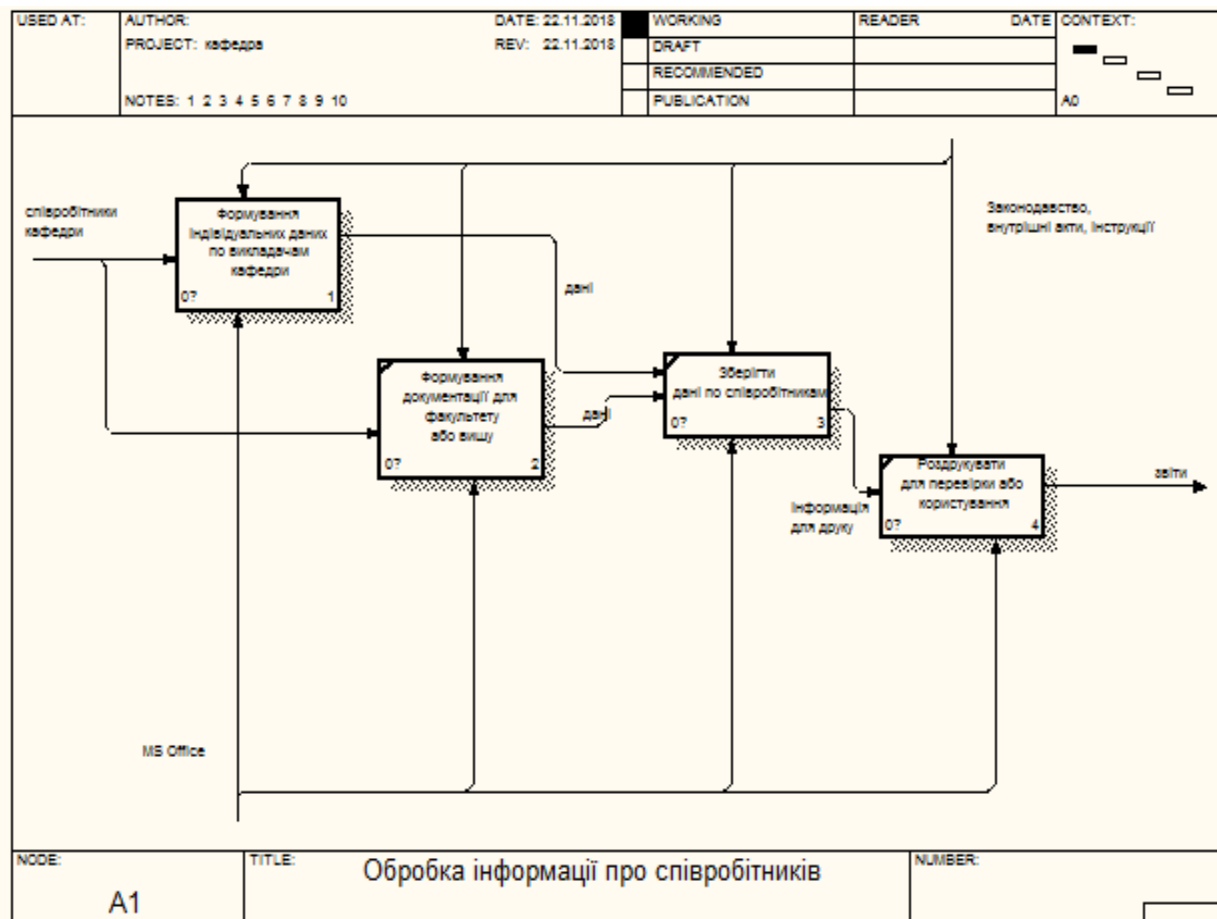


Рисунок 1.3 – Деталізація блоку «Обробка інформації про співробітників»

Деталізація може бути різних рівнів, поки виконувалася деталізація тільки другого рівня, виконаємо деталізацію третього рівня, представивши один з процесів цього блоку.

При цьому виконаємо деталізацію в нотації DFD, цей стандарт оперує термінами процес, сховище даних, сутність та інформаційний потік. Застосовується цей вид діаграм для визначення потоку даних [32].

На рисунку 1.4 представлена деталізація блоку «Формування індивідуальних даних по викладачам кафедри».

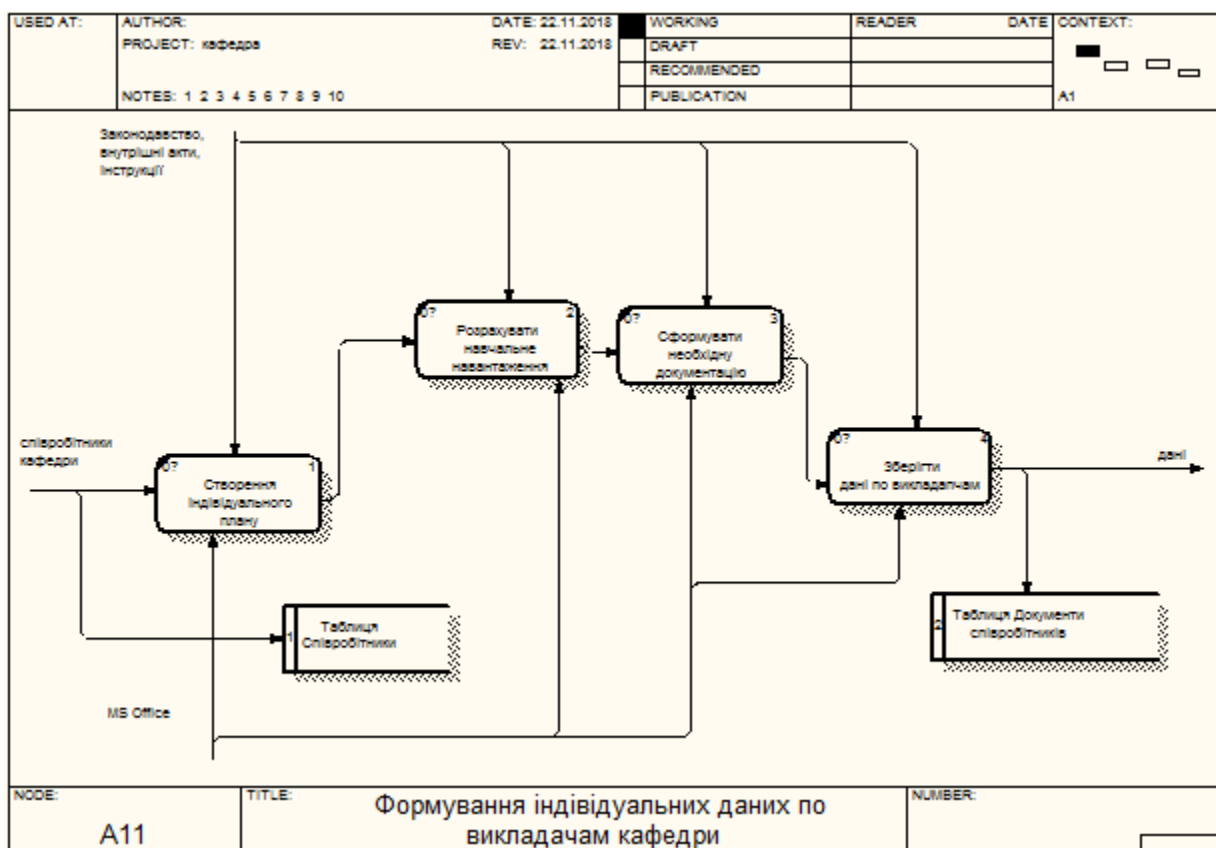


Рисунок 1.4 – Деталізація блоку «Формування індивідуальних даних по викладачам кафедри»

На рисунку представлені наступні процеси:

- створення індивідуального плану;
- розрахувати навчальне навантаження;
- сформувати необхідну документацію;
- зберегти дані по викладачам.

Визначені наступні сховища даних:

- таблиця Співробітники;
- таблиця Документи співробітника.

Окрім того збережені інформаційні потоки, що належать діаграмі ієрархією вище.

На рисунку 1.5 представлена деталізація блоку «Формування документації для факультету або вишу».

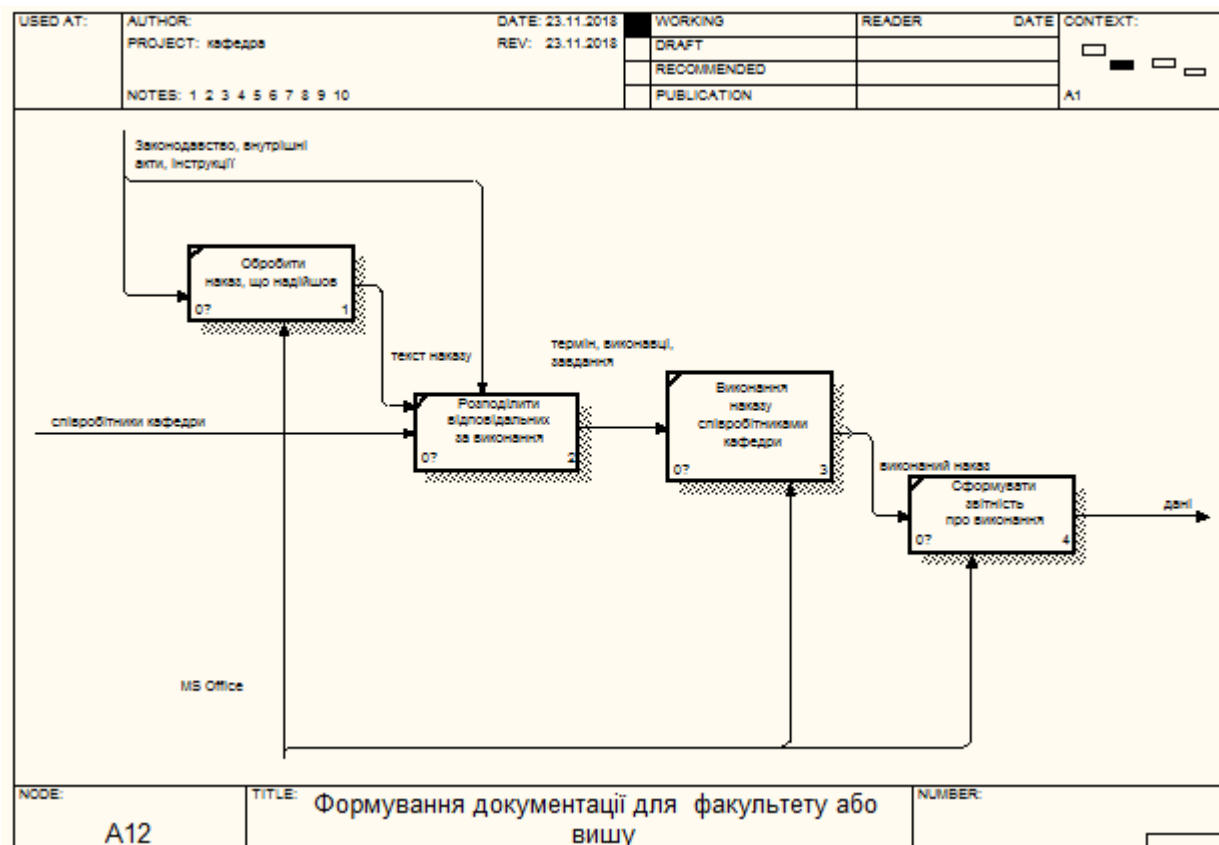


Рисунок 1.5 – Деталізація блоку «Формування документації для факультету або вишу»

Цей процес представлено наступними блоками:

- обробити наказ, що надійшов;
- розподілити відповідальних за виконання наказу;
- виконати наказ співробітникам кафедри;
- сформулювати звітність про виконання.

На рисунку 1.6 представлена деталізація процесу «Формування інформації про студентів кафедри». Цей процес містить наступні блоки:

- вести облік студентів, що навчаються на кафедрі;
- забезпечити необхідними навчальними матеріалами;
- вести облік успішності навчання студентів;
- зберегти інформацію про успішність студентів.

Найбільшу цікавість викликають в цьому процесі два блока «Забезпечити необхідними навчальними матеріалами» та «Вести облік успішності студентів».

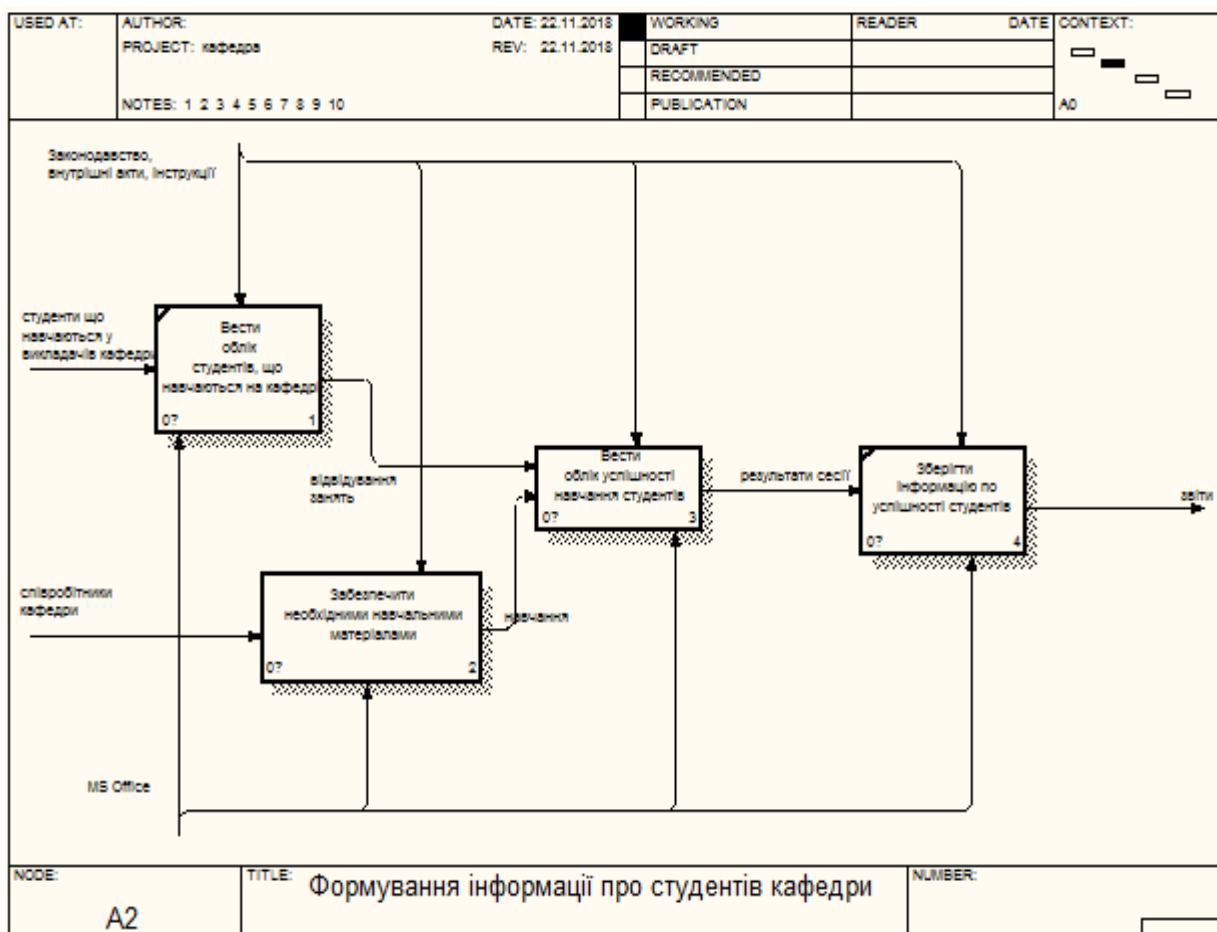


Рисунок 1.6 – Деталізація блоку «Формування інформації про студентів кафедри»

На рисунку 1.7 представлена деталізація блока «Забезпечити необхідними навчальними матеріалами». Цей блок представлений наступними процесами:

- визначити необхідний обсяг для навчання;
- підготувати навчальний матеріал;
- видати матеріал студентам;
- розіслати по електронній пошті;
- проводити навчання.

При цьому визначено наступне сховище даних: таблиця «Навчальні матеріали».

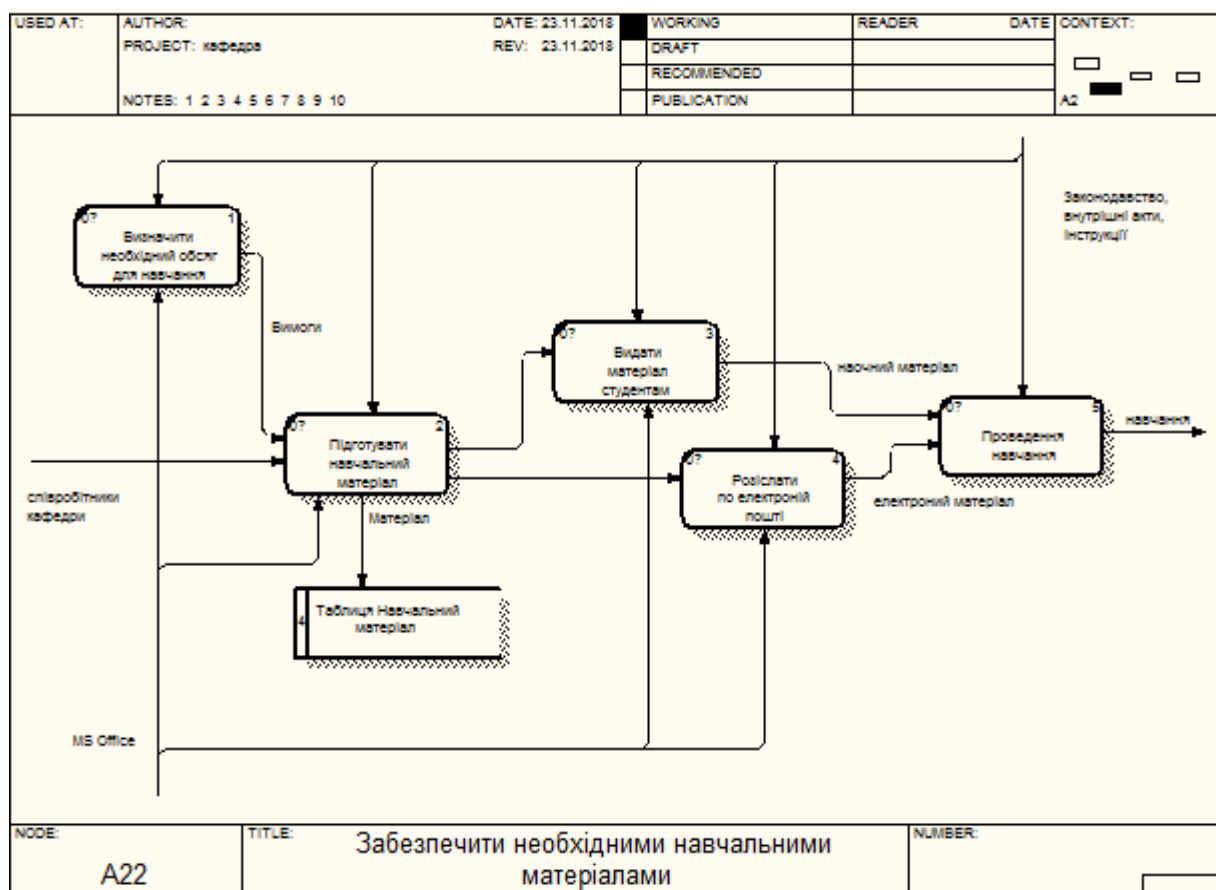


Рисунок 1.7 – Деталізація блоку «Забезпечити необхідними навчальними матеріалами»

Деталізація блоку «Вести облік успішності» представлена на рисунку 1.8.

Процес представлений наступними блоками:

- проведення поточних аудиторних занять;
- виконання завдань впродовж семестру;
- проведення заліків та іспитів;
- виставити оцінки успішності.

Для цього процесу визначено сховище даних – таблиця «Екзаменаційно-залікові відомості».

Взагалі призначення сховищ даних, виявити в процесі аналізу елементи, які необхідно буде реалізовувати в базі даних своєї розробки. Структура та поля цих сховищ будуть визначатися пізніше.

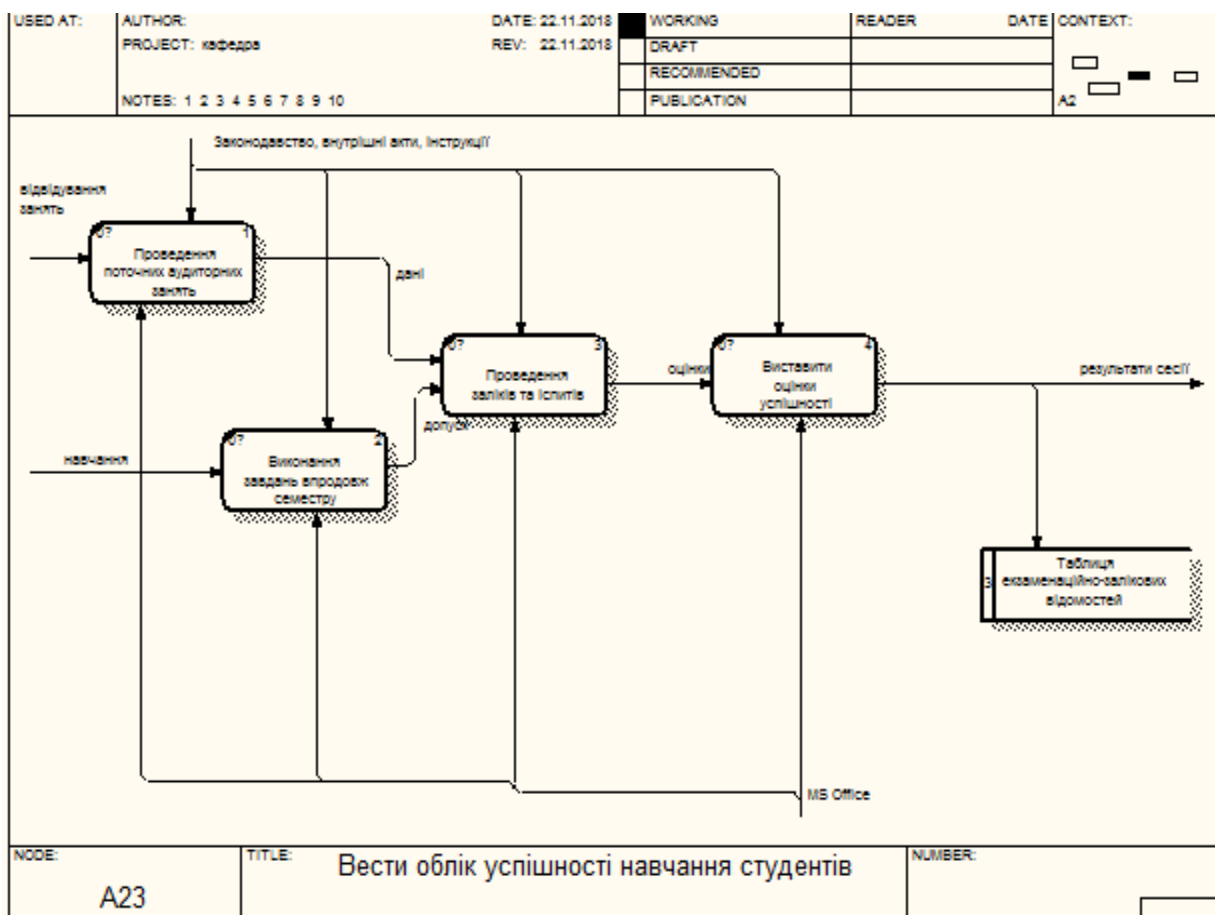


Рисунок 1.8 – Деталізація блока «Вести облік успішності навчання студентів»

На рисунку 1.9 представлена деталізація процесу «Обробка наукових праць кафедри». Наукові праці посідають важливе значення в роботі кафедри – базуючись на них можна впроваджувати нові технології в навчанні, вивчати нові технології або методи. Викладачі, які займаються наукою мають значно вищу кваліфікацію та знання.

Блок представлено наступними процесами:

- написання наукової роботи – це може бути стаття, тези, монографія, підручник, кандидатська або докторська дисертація;
- додати працю до реєстру наукових робіт кафедри;
- сформувати реєстр наукових праць кафедри.

Для цього процесу визначено сховище даних – таблиця Реєстр наукових праць. В даній таблиці будуть зберігатися дані, що відповідають за науковий аспект роботи кафедри.

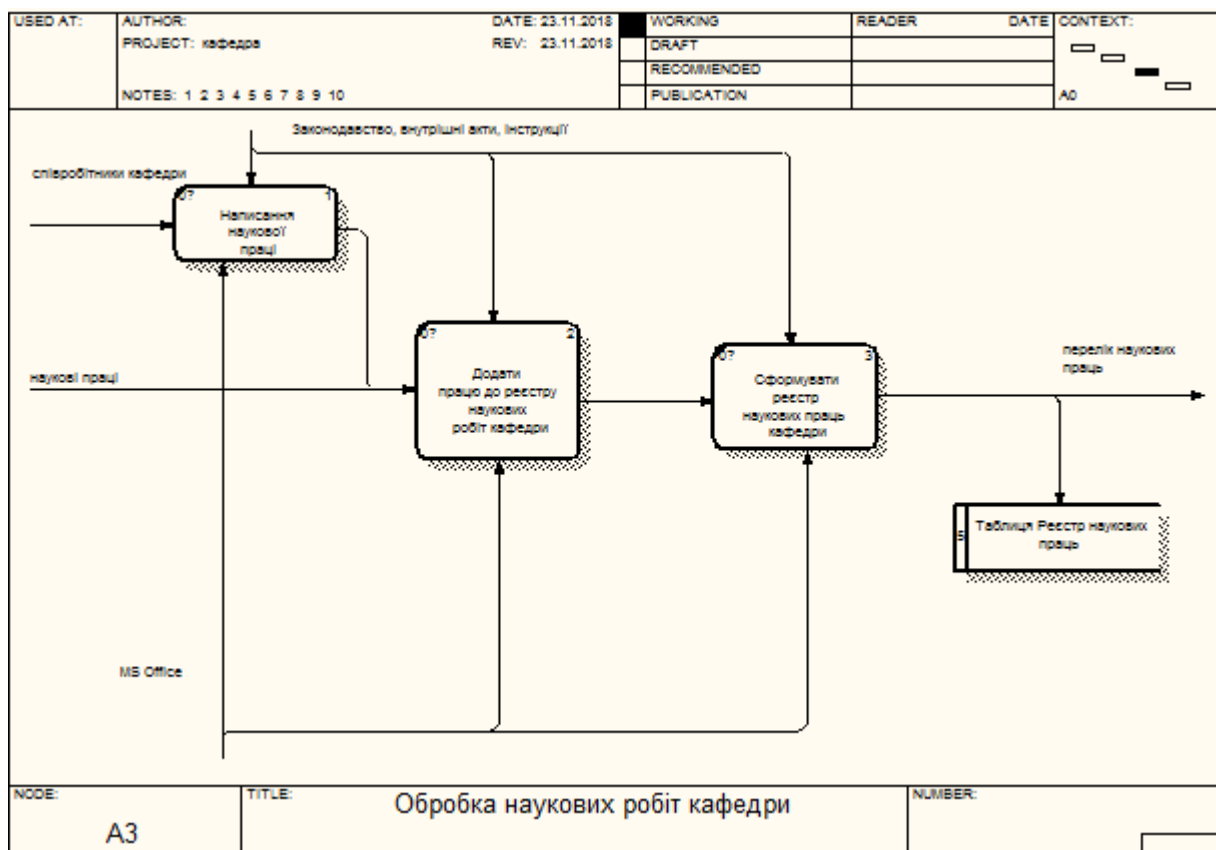


Рисунок 1.9 Деталізація блоку «Обробка наукових робіт кафедри»

Наступний блок «Обробка навчально-методичних матеріалів» представлено на рисунку 1.10. Саме навчально-методичні матеріали забезпечують процес навчання, цей блок повинен бути пов'язаний як з викладачами кафедри, так і студентами (на рисунку 1.7 про цей аспект вже було згадування).

Блок «Обробка навчально-методичних матеріалів» представлені наступними процесами:

- формування навчально-методичного матеріалу у паперовому вигляді;
- формування навчально-методичного матеріалу у електронному вигляді;
- формування реєстру навчально-методичного матеріалу кафедри.

Сховищем даних в цьому блоці виступає раніше створена таблиця Навчальний матеріал.



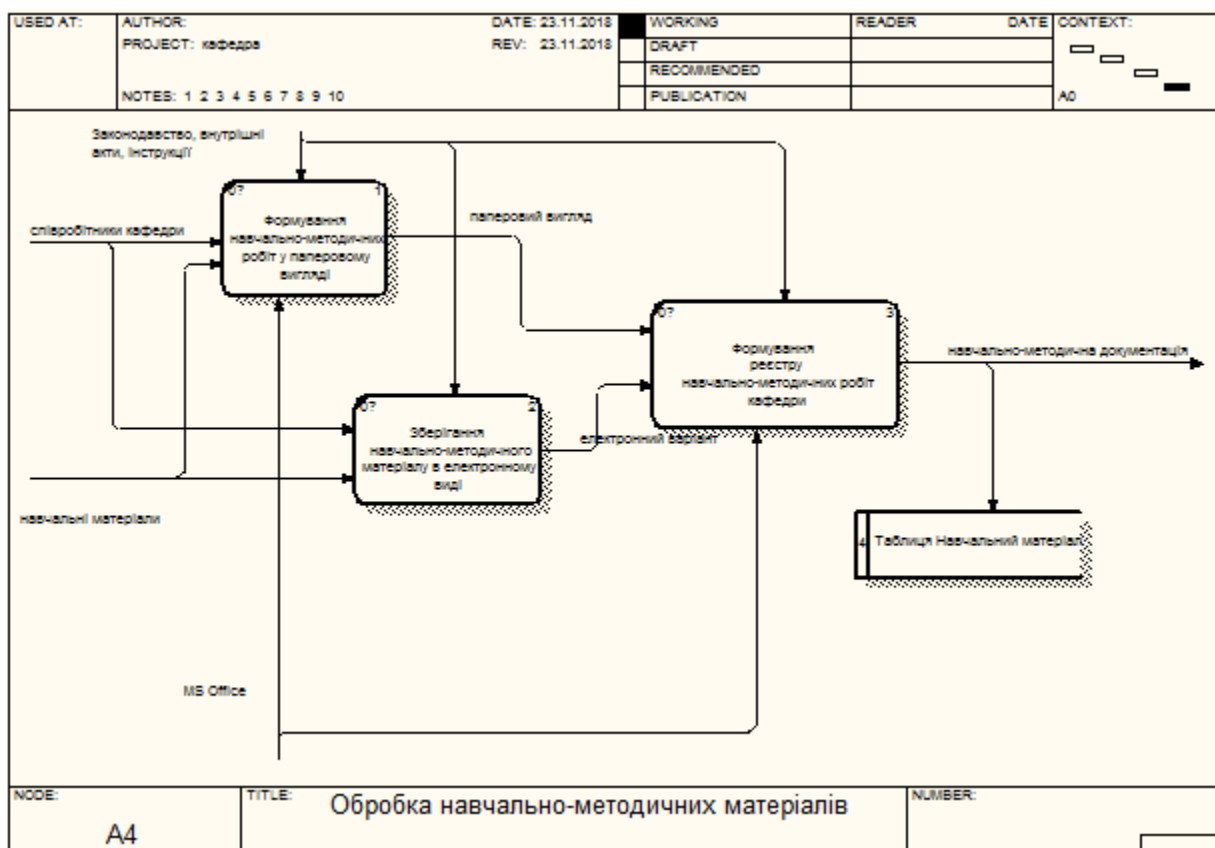


Рисунок 1.10 – Деталізація блоку «Обробка навчально-методичних матеріалів»

Провівши аналіз процесів, можна сформулювати висновок: зберігання матеріалів, що відповідають науковим та педагогічним аспектам роботи кафедри відбувається в більшості в паперовому вигляді, в рідкісних випадках в програмах пакету MS Office, що призводить до втрати, появи помилок та значним часовим витратам [19]. Тому була необхідно розробити систему, яка б допомагала зберігати необхідну інформацію кафедри, дозволяла би її доповнювати, підтримувала б зворотній зв'язок між учасниками процесу навчання.

Головна перевага, яку дає перехід до автоматизованого ведення бази даних – це швидкий пошук необхідних відомостей, надання їх в зручній формі і формування статистичної інформації для подальшого аналізу [23].

Визначимо вимоги до побудови модуля внутрішніх повідомлень веб системи управління науковими та педагогічними аспектами роботи кафедри.

### **1.3 Постановка задачі на розробку модуля внутрішніх повідомлень веб системи управління науковими та педагогічними аспектами роботи кафедри**

Метою роботи є розробка системи управління науковими та педагогічними роботами з модулем обміну інформації для наповнення контентом науково-методичних матеріалів для студентів кафедри.

Створюваний програмний продукт повинен містити базу даних (сховище) з навчально-методичним та науковим матеріалом, необхідним для успішного навчання та наукової роботи студентів, і реалізовувати управління доступом для користувачів системи з різними правами [7].

База даних повинна виконувати стандартні операції, властиві базам даних. У сховищі повинні бути реалізовані наступні функції:

- додавання, редагування, видалення навчально-методичного та наукового матеріалу, який може зберігатися в різних форматах уявлення (текстовий файл, документ Word, відсканований матеріал, малюнки, презентації, електронні таблиці та ін.);
- пошук необхідного матеріалу за різними критеріями (назвою дисципліни, видом наукової парці, ПІБ викладача, № групи студента тощо);
- сортування навчально-методичного та наукового матеріалу за фахом і номеру семестру (курсу);
- зберігання навчально - методичного та наукового матеріалу та архівів попередніх років.

Система управління контентом буде розроблена для трьох категорій користувачів: адміністратора системи, викладача та студента [4].

Адміністратор системи матиме повний доступ до системи та зможе керувати як змістом сховища даних: додавати, редагувати і видаляти файли з навчально - методичним та науковим матеріалом, так і керувати користувачами системи – додавати або вилучати користувачів категорій викладач та студент.

Категорія вчителів буде мати повний доступ до даних сховища системи управління, додавання, редагування, вилучення робіт, що впливають на навчальний та науковий аспект роботи кафедри.

Студенту буде надана можливість користуватися ресурсами системи, здійснювати пошук необхідного навчально-методичного матеріалу за різними критеріями і можливість завантажити необхідний файл, який буде дозволений автором (викладачем).

Повинна бути реалізована система аутентифікації користувачів, яка надавала б доступ відповідно до визначених прав. Система управління контентом повинна бути реалізована з подальшою можливістю впровадження в мережу Інтернет, що дозволить збільшити коло студентів – відвідувачів, надасть можливості використання системи на різних пристроях та місцях.

Окрім обміну необхідного для навчання та наукової діяльності матеріалу, система повинна містити модуль повідомлень, у вигляді листування на електронні скриньки студентів та викладачів.

Система управління контентом призначена для полегшення роботи по адмініструванню бази даних з навчально-методичним та науковими матеріалом викладачів кафедри, що в свою чергу полегшує саму наукову та педагогічну роботу на кафедрі.

До системи управління ставляться такі вимоги:

- мати набір інструментів для управління вмістом сховища (форма для додавання нових документів, форма для редагування обраного навчально-методичного та наукового матеріалу, форма для видалення застарілого або неактуального матеріалу);
- формувати та здійснювати підтримку інтерфейсів надання ресурсів і даних системи для різних категорій користувачів;
- мати систему реєстрації для студентів, що навчаються у викладачів кафедри та мають право на отримання необхідних їм для успішного навчання та наукової праці матеріалів, в якій буде вказано особисті та реєстраційні дані;

- мати засоби пошуку за різними атрибутами навчально-методичеського та наукового матеріалу;
- мати модуль, що забезпечує відправлення та доставку повідомлень між учасниками процесу навчання;
- забезпечувати зберігання і захист навчально-методичного та наукового матеріалу, представленого в різних електронних форматах;
- забезпечувати простоту доступу до сховища навчально-методичного та наукового матеріалу студентам, що навчаються у викладачів кафедри.

Виходячи з функціональних вимог, що пред'являються до системи управління науковими та педагогічними аспектами роботи кафедри, до дипломної роботи були представлені наступні вимоги:

- ознайомитися з етапами створення контенту для надання навчально – методичного та наукового матеріалу;
- ознайомитися з дисциплінами, курсами і видами навчально-методичеського матеріалу, видами науковою роботи, необхідних для навчання та успішної наукової діяльності як викладачів, так і студентів кафедри;
- ознайомиться з необхідними навчальними матеріалами, які повинні бути надані студентам для проходження навчання;
- розробити і реалізувати систему управління інформаційним контентом для подальшого надання навчально-методичного та наукового матеріалу студентам в мережі Інтернет;
- реалізувати сховище навчально - методичного та наукового матеріалу
- організувати систему пошуку за різними критеріями в базі;
- реалізувати засоби для системи управління, які дозволять здійснювати додавання в сховище матеріал, його редагування, в разі зміни навчального плану, і видалення, в разі неактуальності представленого матеріалу;
- передбачити реалізацію сортування матеріалу за фахом, курсу, семестру, видом роботи;

– забезпечити простоту доступу з можливістю перегляду і завантаження необхідного навчального та наукового матеріалу студентам, що навчаються у викладачів кафедри.

#### **1.4 Визначення етапів розробки веб системи управління науковими та педагогічними аспектами роботи кафедри**

У відповідності з різними спостереженнями і дослідженнями визначено, що етапи розробки веб системи розподіляються наступним чином [7]:

стадія 1. Визначення вимог до розробки веб системи управління

стадія 2. Розробка технічного завдання

стадія 3. Проектний ескіз

стадія 4. Програмування веб системи

стадія 5. Налаштування та випробування компонентів

стадія 6. Введення в дію [5]

Уявімо життєвий цикл програмного засобу за допомогою діаграми (див. рис.1.11).

У даній роботі життєвий цикл програмного засобу представлений каскадною моделлю [23]:

- визначення вимог до розробки веб системи управління, впливає Замовник та Проектувальник;
- розробка технічного завдання, впливає Проектувальник;
- розробка проекту, впливає Проектувальник;
- програмування веб системи управління, впливає Програміст;
- налаштування компонентів, впливає Програміст і Замовник;
- введення в дію, впливає Програміст і Замовник.

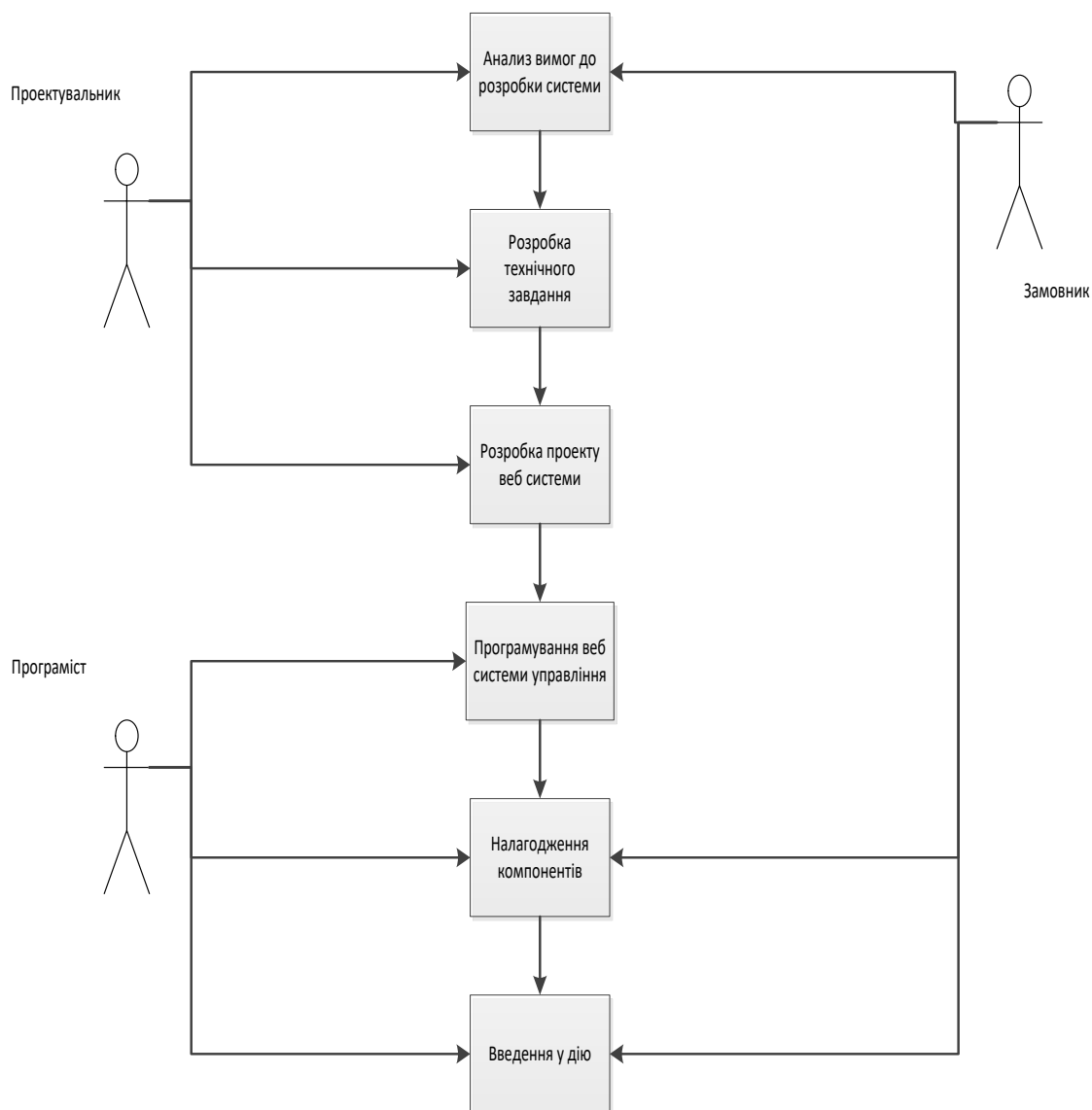


Рисунок 1.11 – Життєвий цикл розробки веб системи управління

Для кожної стадії побудуємо відповідну діаграму. На малюнку 1.12 представлено опис «Визначення вимог до розробки веб системи управління» [19].

На даному малюнку представлені наступні види робіт:

- загальні відомості про проект;
- визначення цілей, призначення і завдань системи;
- проведення досліджень процесів з використанням CASE засобів;
- підготовка вимог до проекту;
- оцінка трудовитрат на проект.

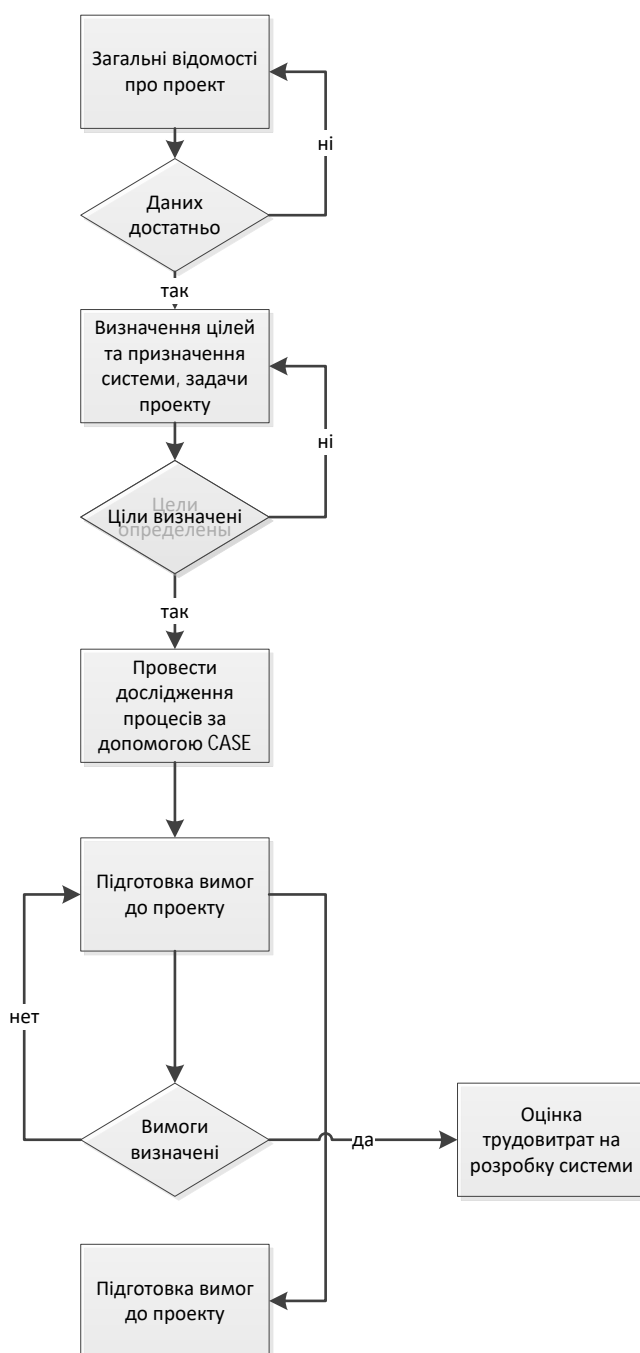


Рисунок 1.12 - Визначення вимог до розробки веб системи управління

Діаграма для наступної стадії «Розробка технічного завдання» представлена на малюнку 1.13. Представлені наступні роботи [7]:

- визначити цілі і завдання;
- призначення системи;
- характеристика автоматизації та опис об'єкта автоматизації;
- визначення вимог до функціональності системи;

- визначення етапів і стадій створення системи;
- вимоги до апаратного забезпечення;
- вимоги до технічного забезпечення;
- вимоги до робочої документації.



Рисунок 1.13 – Розробка технічного завдання

На малюнку 1.14 представлений процес «». Даний процес представлений наступними роботами [23]:

- визначення технологій розробки проекту;



- визначення складу користувачів і їх функціоналу;
- побудова алгоритмічного забезпечення системи;
- вибір платформи;
- проектування логічної моделі БД;
- визначення підпрограм і функцій системи;
- розробка структури програмного забезпечення;
- готове проектне рішення.



Рисунок 1.14 – Розробка проекту веб системи управління

На рисунку 1.15 представлений процес «Програмування веб системи управління педагогічними та науковими аспектами роботи кафедри».

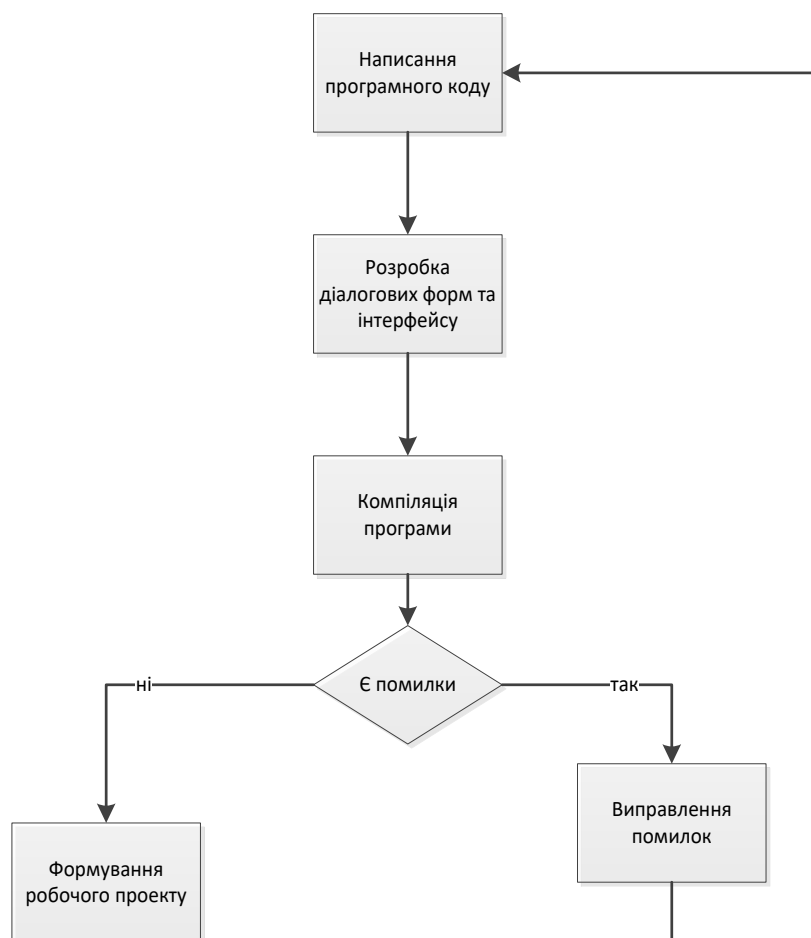


Рисунок 1.15 – Програмування веб системи управління педагогічними та науковими аспектами роботи кафедри

Процес представлений наступними діями [6]:

- написання програмного коду;
- розробка діалогових форм та інтерфейсу;
- компіляція програми;
- виправлення помилок;
- формування робочого проекту.

На рисунку 1.16 представлений процес «Налагодження компонентів». Процес представлений наступними діями:

- налагодження компонентів у відповідності до експлуатаційної документації;
- використання програмних засобів;
- використання технічних засобів.

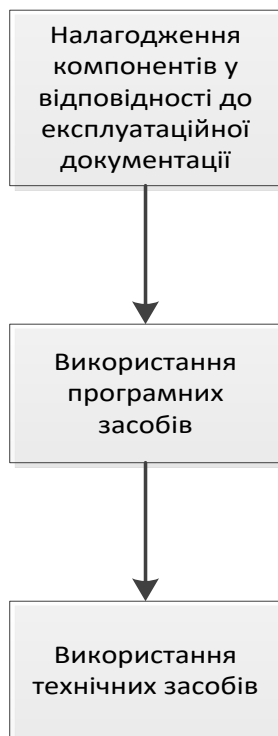


Рисунок 1.16 – Налагодження компонентів

На рисунку 1.17 представлений процес «Введення в дію» [7]. Процес представлений наступними операціями[36]:

- навчання персоналу;
- установка платформи і програми;
- обладнання робочого місця;
- підготовка інформації для бази даних;
- тестування програми;
- усунення несправності і внесення змін в документацію;
- доробка програмного забезпечення програми;
- оформлення акта про впровадження системи.

З існуючих стратегій впровадження веб систем управління для даної розробки найкращим варіантом є паралельна стратегія, це дасть можливість співробітникам

кафедри ще виконувати паперовий облік педагогічним та наукових робіт, поступово здійснюючи перехід на нову веб орієнтовану систему [23].

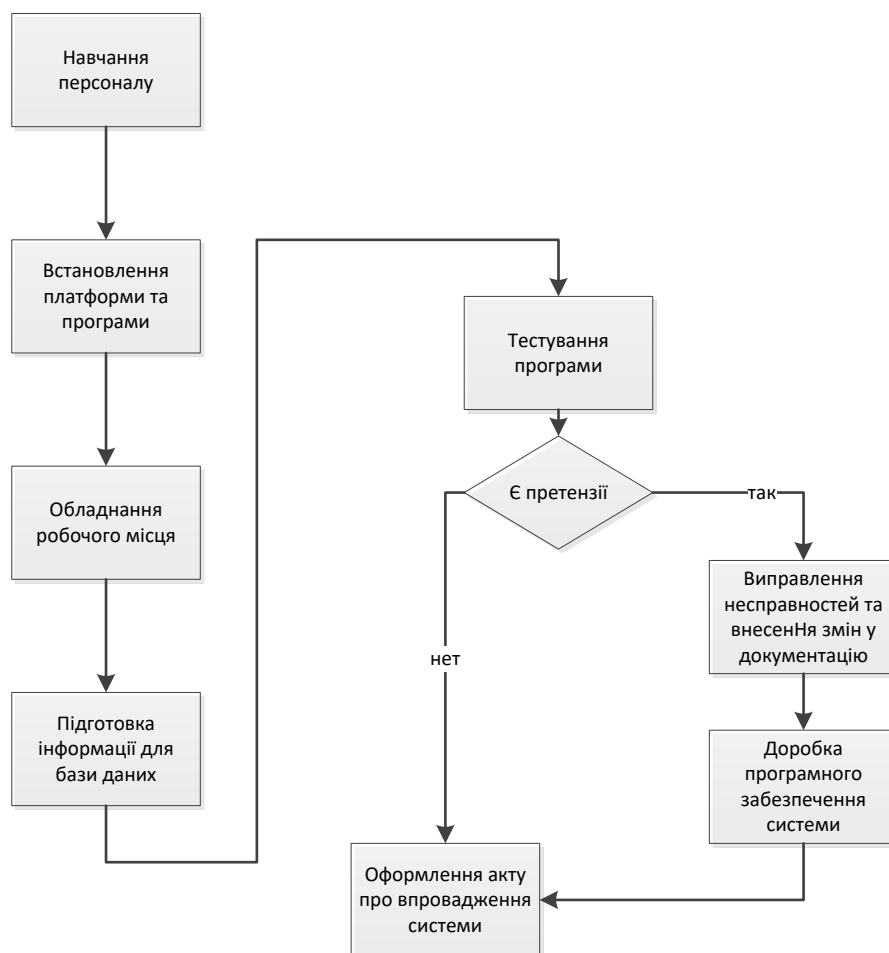


Рисунок 1.17 – Введення у дію

## Висновки по розділу 1

В рамках цього розділу були вирішені наступні питання:

- проведено аналіз діяльності кафедри при виконанні наукових та педагогічних робіт кафедри, аналіз виконано за допомогою стандартів IDEF0, DFD;
- визначені вимоги до побудови веб системи управління науковими та педагогічними аспектами роботи кафедри, сформовано постановку завдання;
- визначені основні етапи розробки веб системи, що значно полегшить виконання завдання.

Наступним кроком роботи буде побудова проекту веб системи управління, спочатку визначимо функції користувачів системи, представимо алгоритмічне забезпечення та обґрунтуємо вибір платформи реалізації.

## **2 РОЗРОБКА ПРОЕКТУ ВЕБ СИСТЕМИ УПРАВЛІННЯ НАУКОВИМИ ТА ПЕДАГОГІЧНИМИ АСПЕКТАМИ РОБОТИ КАФЕДРИ З МОДУЛЕМ ПОВІДОМЛЕНЬ**

### **2.1 Опис функції користувачів системи**

Відповідно до визначених вимог, які висуваються до веб системи управління та згідно раніше розглянутого плану стадії розробки, підчас формування проекту системи необхідно визначити склад користувачів системи та їх функції [4]. Раніше було визначено, що веб система управління буде розроблятися для трьох категорій користувачів:

- адміністратора системи;
- викладачів кафедри;
- студентів, що будуть мати доступ до наукових та педагогічних аспектів роботи кафедри.

Для представлення функції користувачів скористаємося діаграмами прецедентів мови моделювання UML [2].

Відсутність належного спілкування між клієнтом та членами команди проекту часто унеможлиблює правильну ідентифікацію потреб кінцевого користувача. Відмінності в моделях проблеми і рішення, а також необхідність перетворення між ними, представляють собою серйозне джерело неправильного тлумачення, де типові ситуації є: відсутність розуміння потреб користувачів «в цілому», або розуміння «не зовсім» [3]. Популярний метод для вирішення таких завдань є моделюванням прецедентів використання, який дозволяє вказувати функціональність і контекст системи в простій формі, зрозумілий для широкого кола учасників проект.

Дана діаграма базується на термінах: актор та варіант використання [12].

Актором називають когось (або щось), що взаємодіє з системою ззовні.

Актор представляє потенційного користувача системи, що розглядається як ціле, або користувача лише певної частини системи (підсистеми або навіть одного

класу). Потенційним користувачем є будь-яка сутність із оточення частини тієї системи, яка стосується відповідної функціональності [2].

Актор може делегувати завдання до системи (наприклад, підтвердження отримання заяви, розміщення замовлення, відправлення повідомлення на об'єкт) та / або співпрацювати з системою під час виконання цього завдання [14].

Актор повинен мати унікальне ім'я.

Актор є головною причиною формування випадків використання, що означає, що він є ініціатором, що викликає події, щоб запустити випадок використання, а також відправник і одержувач даних до / від випадку використання [15].

Актор може взаємодіяти з системою різними (відповідними) способами [3]. Кожен з цих методів називається варіантом використання і являє собою послідовність дій, реалізованих системою, в результаті чого спостережувані результати доставляються акторові [35].

Дія – це операція (виконується повністю або взагалі не виконується взагалі), яка реалізується системою у відповідь або на сигнал від актора, або на подію, пов'язану з проходженням часу ід на потік подій між актором і системою [15]. Послідовності дій згруповані у варіанти використання.

Послідовність дій, реалізованих системою – це важлива компонента для визначення меж системи та визначення сфери її відповідальності [14]. Тут чітко визначено, що робить система (вона відрізняється від дій зовнішнього світу і чітко відокремлена від них).

На рисунку 2.1 представимо діаграму варіантів використання для категорії користувачів «Адміністратор веб системи управління».

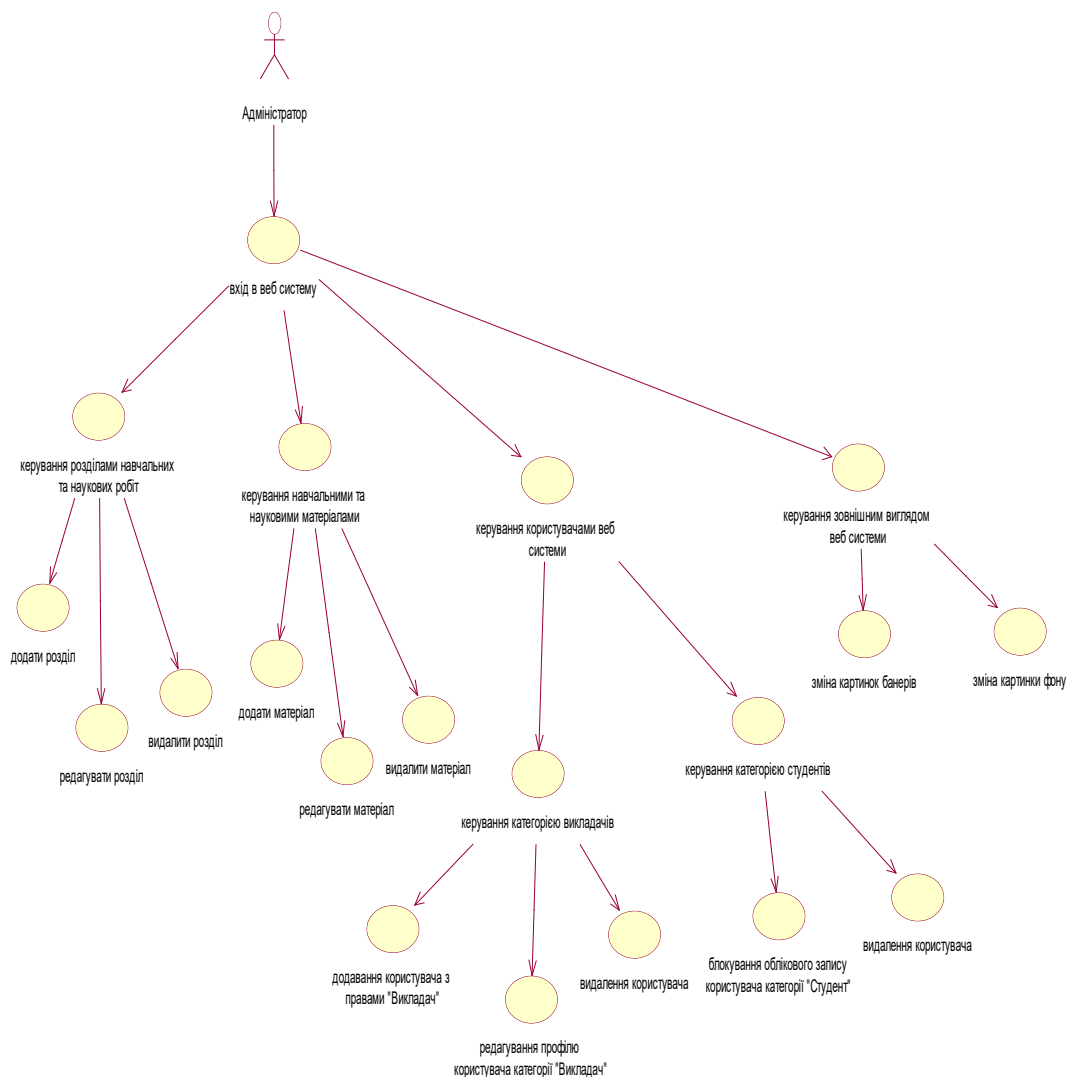


Рисунок 2.1 – Діаграма варіантів використання для категорії користувачів  
«Адміністратор веб системи управління»

Із представленого рисунку видно які функції будуть притаманні цій категорії користувачів, представимо їх.

1. Керування розділами навчальних та наукових матеріалів, в цей вид робіт входить:

- додати розділ;
- редагувати розділ;
- видалити розділ.

2. Керування навчальними та науковими роботами:



- додати матеріал;
- редагувати матеріал;
- видалити матеріал.

3. Керування користувачами веб системи, цей варіант використання породжує наступні дії:

– керування категорією викладачів, цей варіант використання також породжує деякі дії:

- додавання користувача з правами «Викладач»;
- редагування профілю користувача з правами «Викладач»;
- видалення користувача;
- керування категорією студентів:
  - блокування облікового запису користувача категорії «Студент»;
  - видалення користувача.

4. Керування зовнішнім виглядом веб системи управління:

- зміна картинок банерів;
- зміна картинок фону.

На рисунку 2.2 представлена діаграма варіантів використання для категорії користувачів «Викладач». Головною функцією викладачів є додавання матеріалу, що забезпечує роботу кафедри, а також підтримка спілкування з учасниками навчання – студентами або іншими викладачами.

Представимо яким чином буде додаватися матеріал в веб системі управління:

- вказати викладача (або обрати із переліку);
- вказати тип роботи;
- вказати дисципліну;
- вказати спеціальність;
- вказати курс;
- вказати назву матеріалу;
- заповнити короткий опис матеріалу;
- вказати шлях до файлу.

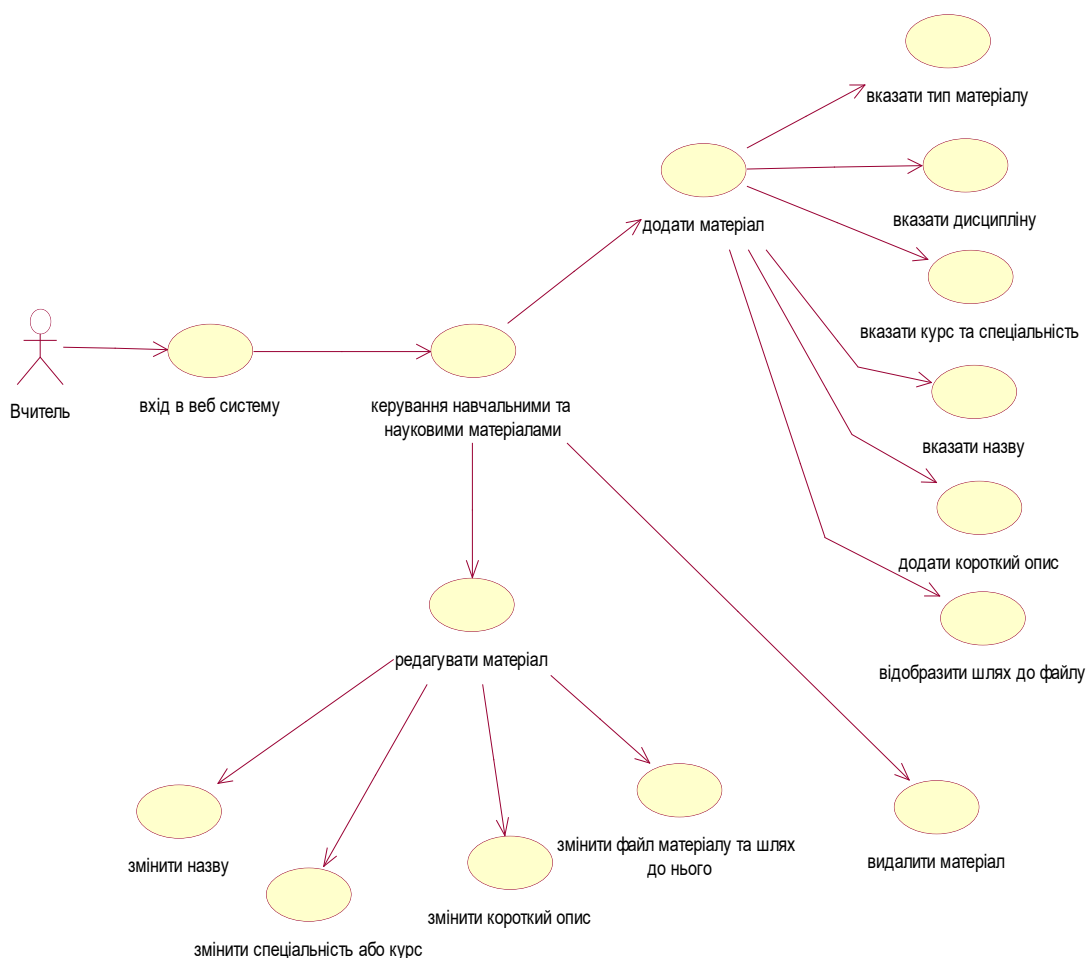


Рисунок 2.2 – Діаграма варіантів використання для категорії «Викладач»

На рисунку 2.3 представлена діаграма варіантів використання для категорії користувачів «Студент». Для користувача категорії «Студент» будуть доступні наступні операції:

1. Реєстрація у веб системі управління, при реєстрації необхідно виконати такі дії:

- вказати ім'я та прізвище студента;
- пароль та його підтвердження.

2. Робота з каталогом матеріалів, студенту буде доступна функція пошуку, пошук може відбуватися за наступними критеріями:

- за типом робіт;
- пошук за викладачем;

- пошук за алфавітом;
- пошук за спеціальністю або групою.

### 3. Завантаження знайденого матеріалу.

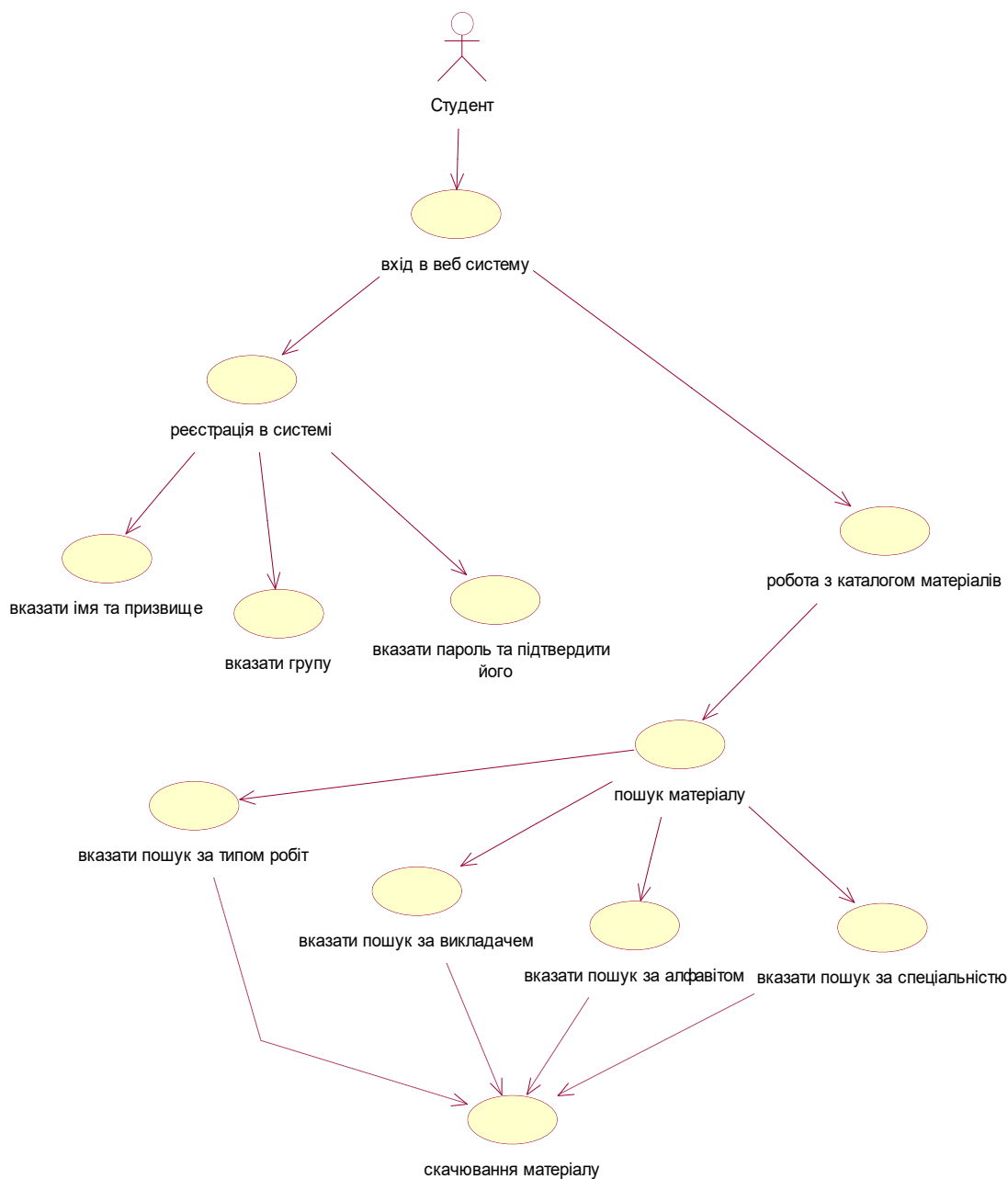


Рисунок 2.3 – Діаграма варіантів використання для категорії користувачів  
«Студент»

Визначивши користувачів системи та їх функції необхідно виконати побудову алгоритмічного забезпечення веб системи управління науковими та педагогічними

аспектами роботи кафедри, в подальшому це дасть змогу визначити основні об'єкти та принципи їх роботи, сформує уявлення механізму взаємодії між елементами системи [7].

## **2.2 Розробка алгоритмічного забезпечення веб системи управління**

Проектована веб система призначена для підтримки виконання наукової та педагогічної роботи на кафедрі з можливістю обміну повідомлень між учасниками процесу навчання. Система буде являти собою сховище електронних методичних та наукових матеріалів, які будуть доступні користувачам контенту [4]. Наповнення здійснює адміністратор або викладач. Загальний вигляд функціонування веб системи представлений на рисунку 2.4.

База електронного навчально - методичного матеріалу буде розміщена в мережі інтернет, студенти будуть здійснювати вхід, вводячи логін і пароль, отриманий при реєстрації в системі, після входу надаються ресурси бази. Доступ до ресурсів проводиться через каталог, система пошуку дозволить знаходити матеріал за різними критеріями (назва дисципліни, ПІБ викладача, № групи студента) [19].

Велику роль в наповненні і функціонуванні контенту виконує система управління [22]. Система управління буде виконувати наступні функції:

- містити інструментальні засоби для керування вмістом сховища (форму для додавання нових документів, форму для редагування обраного навчально – методичного та наукового матеріалу, форму для видалення застарілого або неактуального матеріалу);
- розміщувати навчально - методичний та науковий матеріал кафедри в базі системи;
- містити систему реєстрації для студентів, що навчаються у викладачів кафедри;
- формувати реєстраційні дані на підставі введених особистих даних (пароль буде формуватися по групі користувача і за алгоритмом випадкової генерації);

- відстежувати номер групи користувача, по введеному паролю, це потрібно для здійснення пошуку по групі та для подальшої аутентифікації;
- здійснювати підтримку категорій користувачів – адміністратора, викладача і студента;
- формувати і здійснювати підтримку інтерфейсів надання ресурсів і даних системи для різних категорій користувачів;
- містити засоби пошуку за кількома критеріями:
  - за назвою дисципліни;
  - за типом матеріалу (контрольна, курсової, наукова праця і т.п.);
  - за номером групи студента (номер групи дозволить визначити спеціальність і номер курсу і видати відповідний список дисциплін);
  - на прізвище викладача;
- забезпечувати зберігання і захист навчально-методичного та наукового матеріалу, представленого в різних електронних форматах;
- забезпечувати простоту доступу до сховища навчально-методичного та наукового матеріалу студентам, що навчаються у викладачів кафедри та мають права на користування матеріалами;
- мати можливість підтримки спілкування між учасниками навчального процесу – надсилання повідомлень на електронну скриньку.

Визначивши функції системи управління, можна зробити висновок, що система, являє собою комплекс підсистем, визначимо основні підсистеми:

- підсистема адміністратора;
- підсистема викладача;
- підсистема користувача;
- підсистема підтримки бази даних;
- підсистема пошуку;
- підсистема реєстрації та формування реєстраційних даних;
- підсистема повідомлень.

Загальний вигляд системи управління представлений на рисунку 2.5.

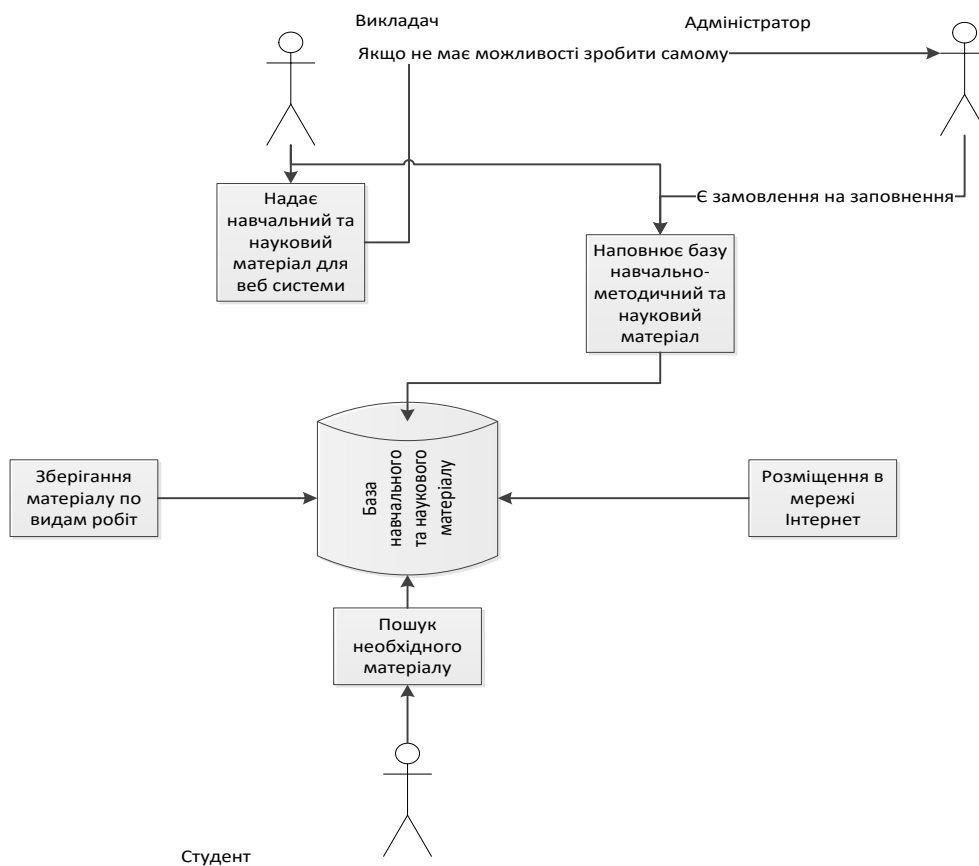


Рисунок 2.4 – Функціонування веб системи управління

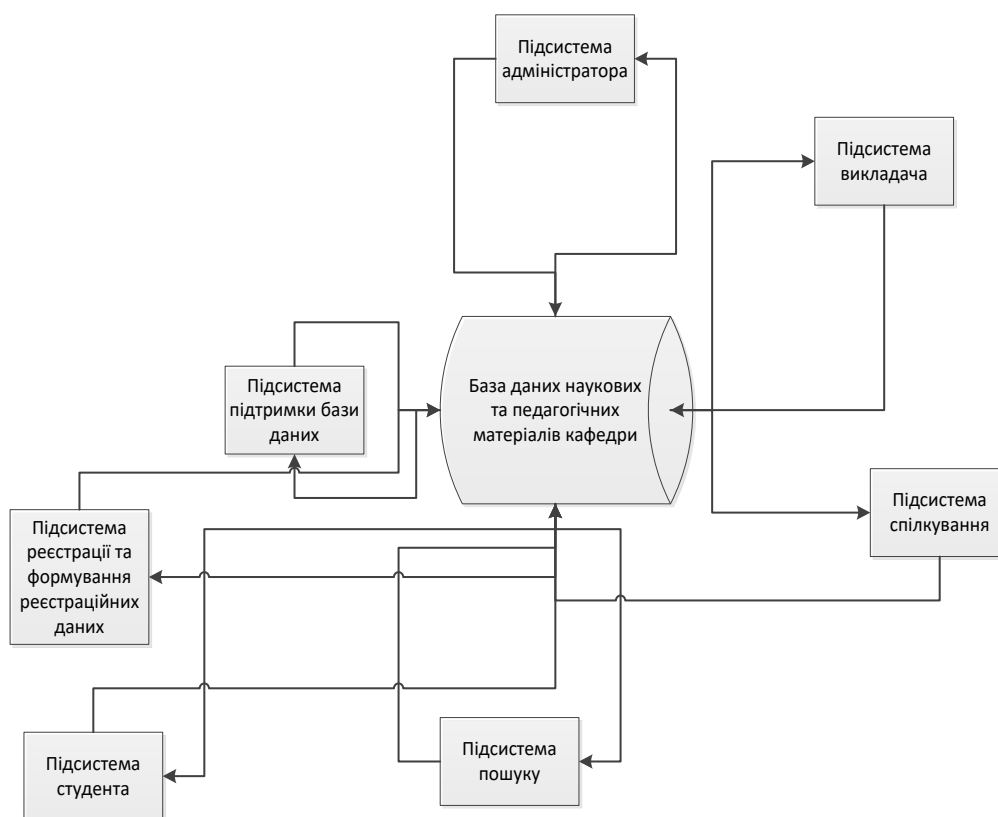


Рисунок 2.5 – Загальний вигляд веб системи управління

Підсистеми взаємодіють між собою і використовують ресурси бази.

Наступним кроком проектування буде розробка структур підсистем [7].

Підсистема адміністратора дозволяє управляти ресурсами бази. Ця підсистема дозволить користувачеві з правами «адміністратор» виконувати такі дії:

- додавати навчально - методичний та науковий матеріал;
- редагувати матеріал;
- видаляти застарілі і неактуальні матеріали з бази;
- додавати новини та оголошення актуальні для студентів заочної форми навчання;
- керувати даними користувачів інших категорій.

Робота підсистеми адміністратора представлена на рисунку 2.6.

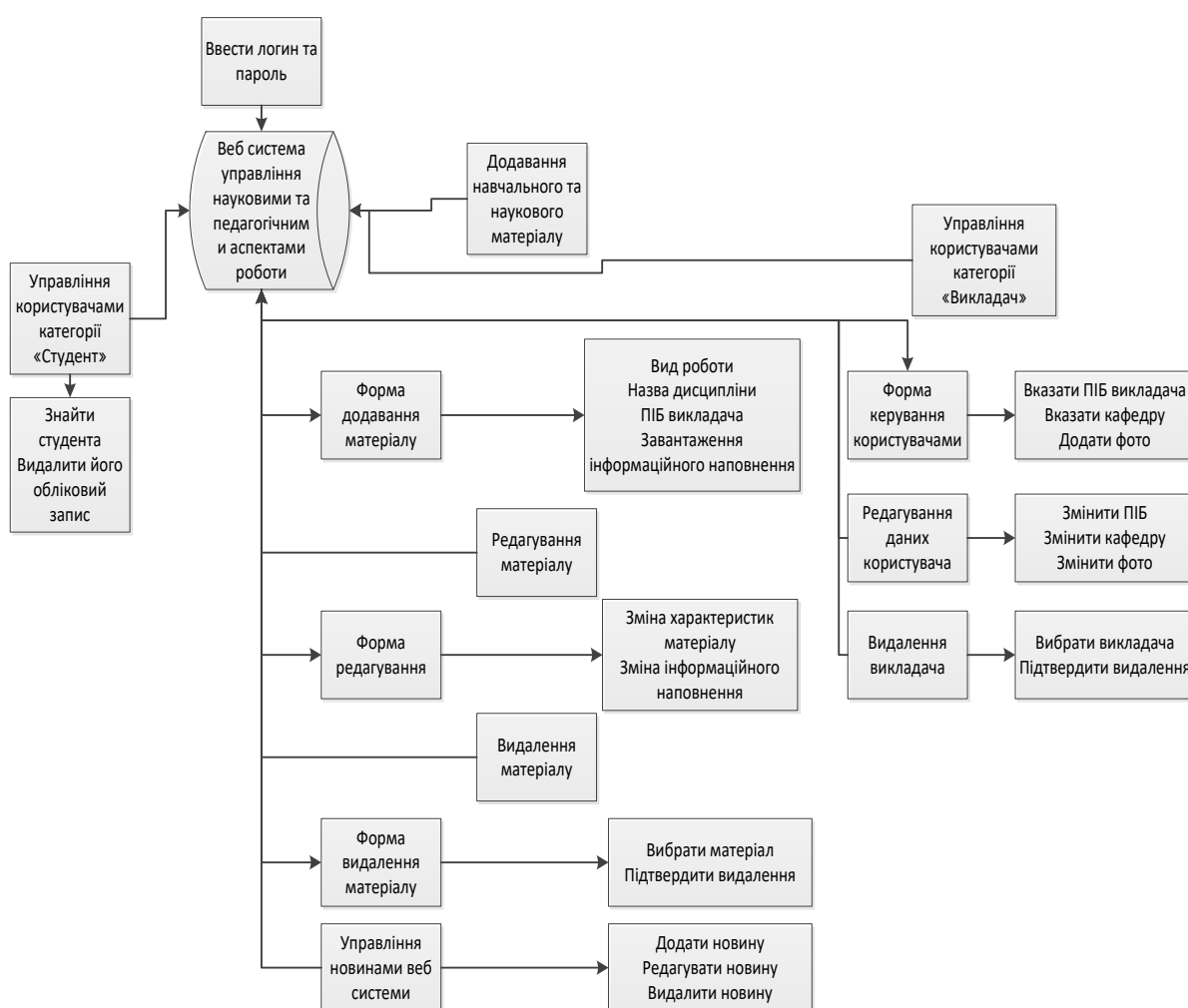


Рисунок 2.6 – Робота підсистеми адміністратора

Додавання в базу навчальних та наукових матеріалів повинно здійснюватися в діалоговому вікні, з використанням діалогу з адміністратором системи[37]. У цій діалоговій формі вказується наступна інформація:

- назва дисципліни;
- список спеціальностей, для яких читається дана дисципліна (вказується шифр і назва);
- кваліфікаційний рівень (бакалавр, спеціаліст, магістр);
- форма звітності з дисципліни (залік / іспит, наявність курсового проекту);
- порядок вибору номера варіанта контрольної роботи;
- безпосередньо сам зміст навчального матеріалу.

При редагуванні матеріалу видається відповідна форма, редагувати можна розділи інформації по навчально-методичному та науковому матеріалу або безпосередньо сам матеріал.

Видалення матеріалу буде здійснюватися в такий спосіб:

- вибір матеріалу;
- підтвердження видалення.

Керування користувачами веб системи управління відбувається наступним чином, при додаванні користувача категорії «Викладач» видається форма в якій необхідно заповнити наступні поля:

- ПІБ;
- кафедра;
- фото.

Якщо необхідно змінити дані по викладачам, викликається відповідна форма, яка надає змогу змінити значення в вище зазначених полях.

При видаленні викладача, необхідно його обрати з переліку викладачів, що зареєстровано в веб системі та підтвердити видалення запису.

При керуванні обліковими записами студентів адміністратор може заблокувати або видалити обліковий запис студента, знайшовши його по номеру групи або за прізвищу.



Адміністратору також зможе розміщувати актуальні новини та оголошення для студентів, що навчаються у викладачів кафедри.

На рисунку 2.7 представлено роботу підсистеми викладача. Викладач має можливість розміщення наукових та педагогічних робіт, він в цьому наслідує властивості адміністратора веб системи, тобто додавати, редагувати та видаляти матеріали, заповнюючи відповідні поля в формах, що надаватиме система управління.



Рисунок 2.7 – Робота підсистеми викладача

Підсистема користувача дозволить студенту користуватися ресурсами веб системи управління педагогічними та науковими аспектами роботи кафедри, здійснювати пошук необхідного навчально-методичного та наукового матеріалу за різними критеріями, можливість завантажити необхідний файл та формувати повідомлення для інших учасників процесу навчання – студентам та викладачам. Принцип функціонування цієї підсистеми представлений на рисунку 2.8.

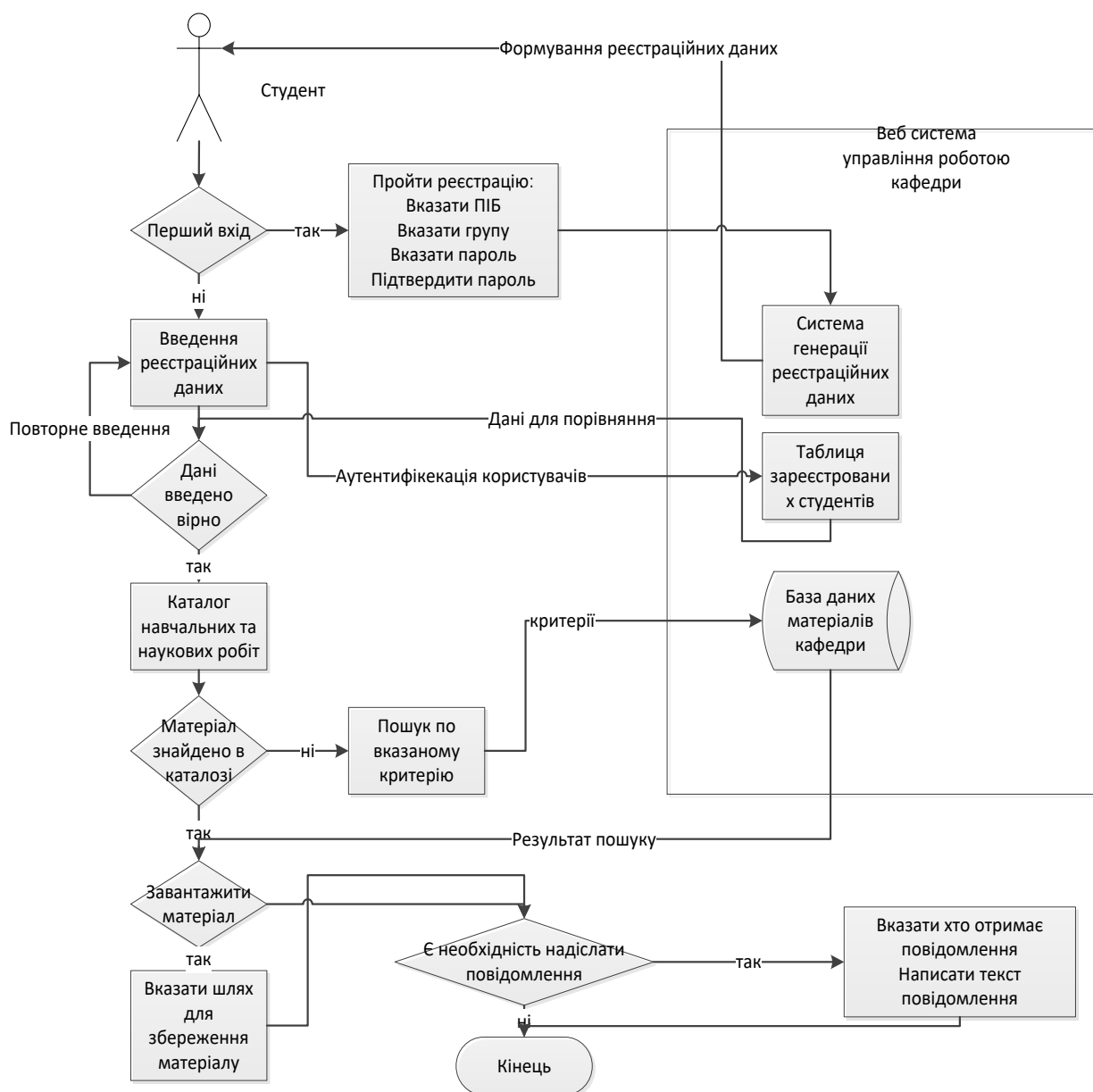


Рисунок 2.8 – Робота підсистеми студента

Для того, щоб підсистема пошуку дозволила студенту знаходити необхідний навчально-методичний матеріал швидше, необхідно буде програмно реалізувати пошук за наступними критеріями[41]:

- за назвою дисципліни;
- за типом матеріалу (контрольна, курсової і т.п.);
- за номером групи студента (номер групи дозволить визначити спеціальність і номер курсу і видати відповідний список дисциплін);
- на прізвище викладача.

Робота підсистеми пошуку представлена на рисунку 2.9.

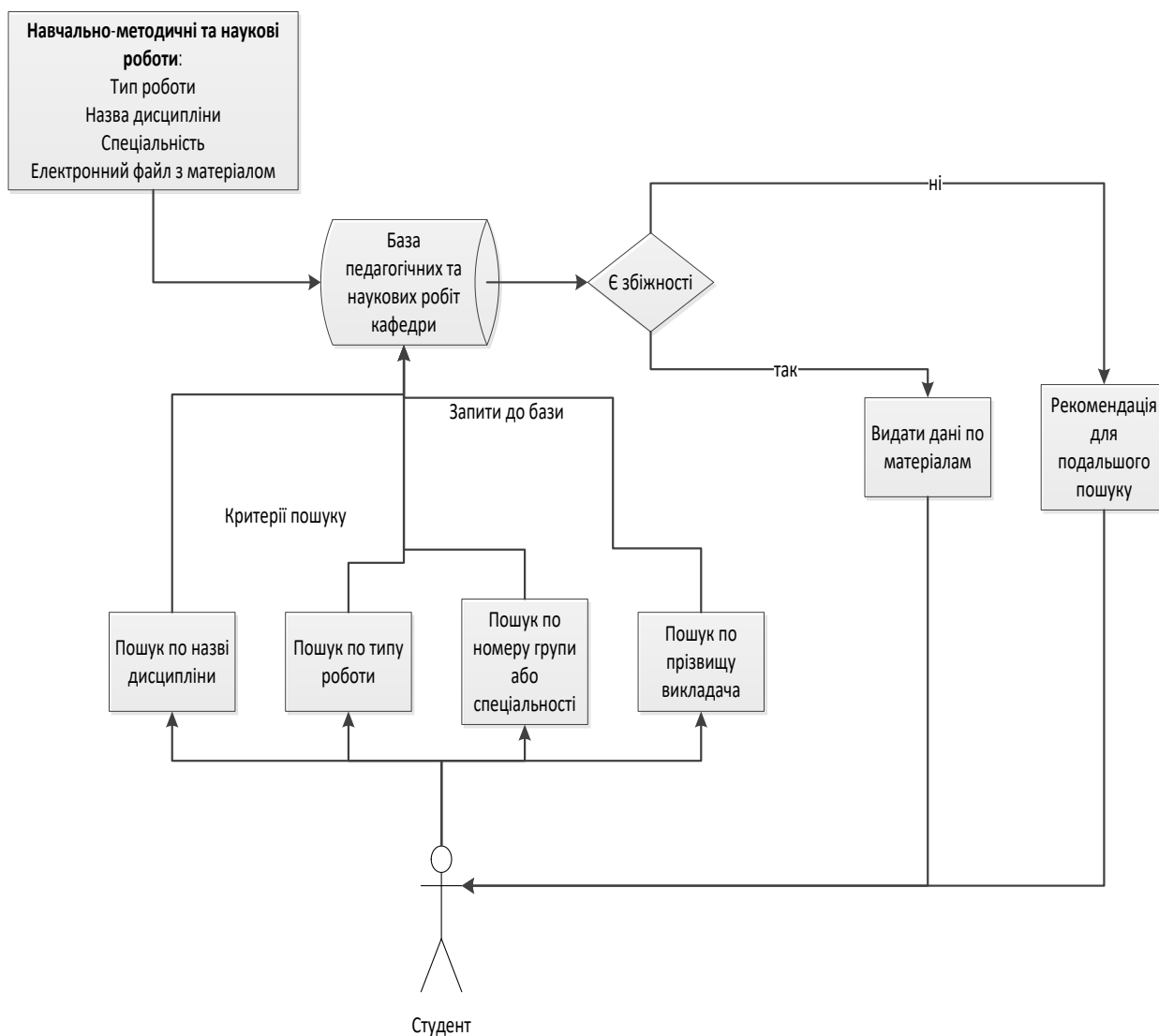


Рисунок 2.9 – Робота підсистеми пошуку

Для зручності пошуку дисципліна курсу буде розбита на компоненти:

- тематика лекцій;
- контрольні роботи та курсовий проект (у разі наявності);
- питання до іспиту чи заліку;

– список літератури.

Підсистема реєстрації і формування реєстраційних даних необхідна для розподілу ресурсів бази електронних навчальних матеріалів між користувачами системи[36]. Реєстрація необхідна також для управління та впорядкування списків користувачів системи. Реєстрація буде здійснюватися при першому вході в систему, будуть заповнюватися поля реєстраційної форми, в ній будуть вказані: прізвище, ім'я, по батькові, логін, e-mail, група. По групі буде формуватися пароль, підсистема сформує повідомлення електронної пошти з логіном і створеним паролем і відправить на поштову скриньку користувача, отримані реєстраційні дані користувач введе і буде зареєстрований в базі студентів, після цього він буде мати доступ до ресурсів системи[39]. На рисунку 2.10 представлена робота підсистеми реєстрації та формування реєстраційних даних.

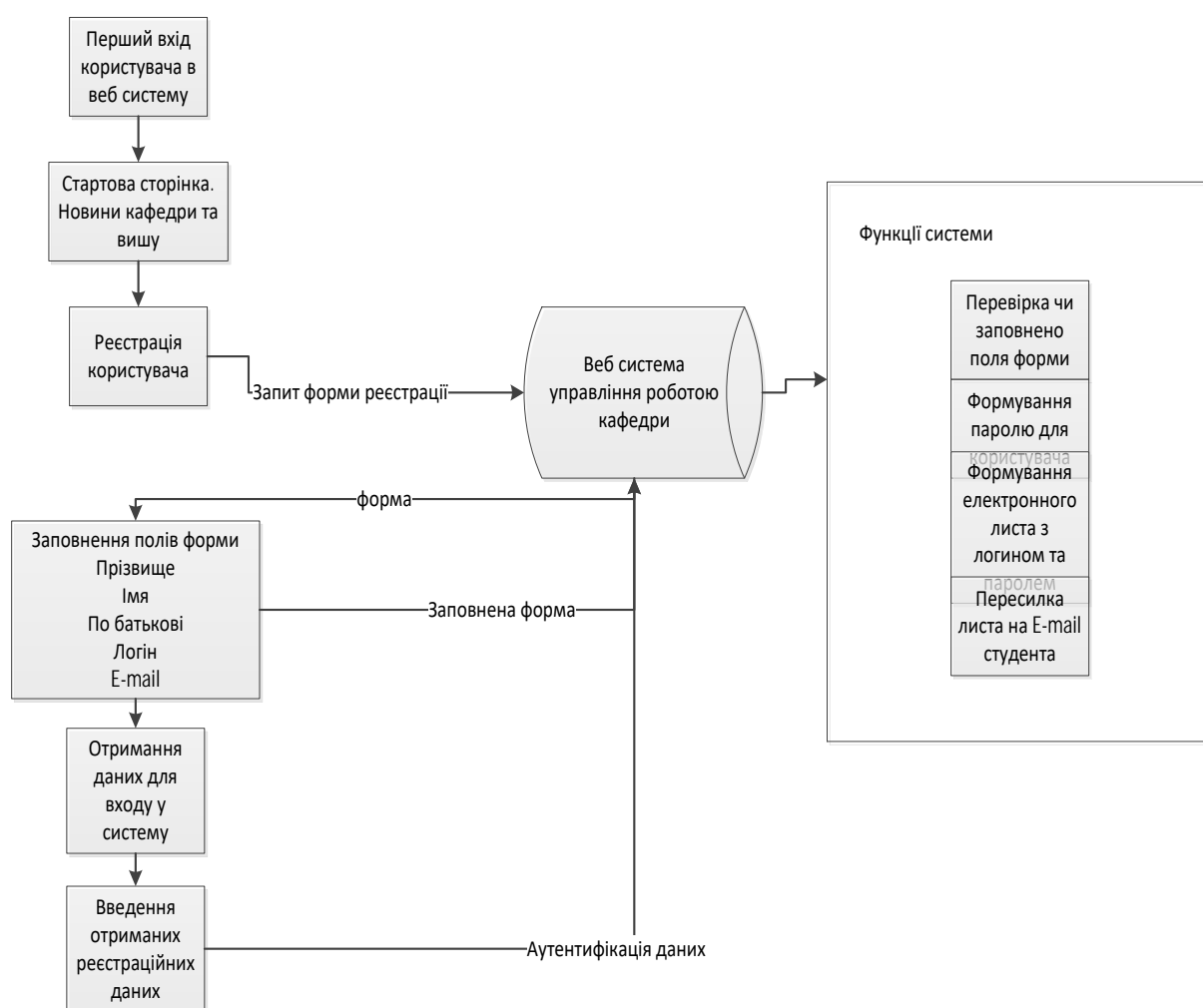


Рисунок 2.10 – Робота підсистеми реєстрації

## 2.3 Вибір платформи

При створенні веб додатків розробник стикається з двома напрямками по розробці веб додатки [4]. Перший напрямок створення сайту класичним методом, тобто написання коду сайту на одному або декількох мовах веб програмування. Залежно від вимог до сайту, розробка може здійснюватися в простих або в візуальних редакторах HTML і CSS [10,34]. У разі, якщо необхідно реалізувати простий статичний сайт, то досить використання «зв'язки» HTML і CSS, з можливим включенням, в деяких випадках, Javascript. Для створення більш функціональних сайтів з можливістю динамічного оновлення має сенс використання таких технологій як PHP, ASP.NET і т.д [1,4].

Для створення сайтів з використанням мови PHP програмний код можна писати в звичайному блокноті операційної системи Windows для запуску знадобиться установка веб сервера, наприклад Денвер [24].

Одним з недоліків ручної розробки є те, що дизайн сайту також необхідно створювати вручну за допомогою різних програмних засобів, для цих цілей можуть застосовуватися будь-які графічні редактори [9].

Другий напрямок по розробці сайтів пов'язаний з використанням методів автоматизованого створення за допомогою систем управління контентом, які називаються спеціальними конструкторами сайтів [22].

CMS – це онлайн-системи, які дають можливість з готового типового набору модулів і компонентів розробити структуру сайту і відразу розмістити його на хостингу. Найбільш популярними CMS є WordPress, Joomla та ін. Необхідно відзначити, що методи створення сайтів з використанням CMS, на даний момент, є найбільш популярними, це пояснюється тим, що не потрібно великих знань програмування у розробників, і надає більш гнучку, готову програмну оболонку , яку користувач може змінити і налаштувати [27].

Розглянемо найбільш популярні платформи для створення веб додатків. Найбільш популярними платформами вважаються наступні:

- Joomla;

- WordPress;
- Drupal;
- TYPO3.

Розглянемо їх властивості, щоб визначити на якій платформі краще виконувати подальшу розробку.

WordPress – це найпопулярніша CMS у світі[38]. Цей скрипт у своєму призначенні більше блог, ніж типовий CMS. Ця CMS є сценарієм Open Source та був розроблений як інструмент для створення блогів, але його також можна успішно використати для розробки простих веб-сайтів [27,40].

Перевагами WordPress є

1. Простота у використанні. WordPress не вимагає жодних знань про PHP або HTML. WordPress має попередньо встановлені функції для плагінів і шаблонів, при цьому встановлення цих плагінів та шаблонів відрізняється простотою.

2. Широке використання. WordPress має одну з найбільших Інтернет спільнот (англійською мовою). Існує також технічна підтримка на різних мовах.

3. Наявність великої кількості доступних плагінів. Сценарій має кілька тисяч плагінів, доступних для створення веб сторінки. Завдяки їм, WordPress був визнаний CMS, а не просто сценарієм блогів. Переважна більшість плагінів доступна безкоштовно.

4. Наявність великої кількості шаблонів. На сторінці сценарію розробник може переглядати тисячі графічних шаблонів, які дозволяють змінювати зовнішній вигляд сторінки.

5. Управління меню. WordPress має великі можливості керування меню, які можна перетворити, щоб включити категорії, сторінки, зображення тощо.

6. Спеціальні поля [28].

Розробник має можливість додавати поля в форми або використовувати плагін. Завдяки цьому записи в блозі або підсторінках матимуть додаткові мітки, класифікації чи описи.

Недоліками WordPress вважають [27].

1. Модифікація вимагає знань PHP. Приклад видалення поля дат вимагає орієнтації на PHP. Для більшості цього виду модифікації потрібна знання цієї мови.

2. Для зміни графіки потрібні знання CSS та HTML. Хоча сценарій має багато шаблонів для створення сторінки, вони в основному схожі один на одного. Проте для створення унікального вигляду необхідні знання CSS та HTML.

3. Плагіни та продуктивність. WordPress вимагає встановлення багатьох плагінів, щоб стати конкурентом з точки зору функціональності Joomla або Drupal. Плагіни впливають на ефективність цього сценарію. Ще однією проблемою є вибір правильного плагіна.

4. Безпека PHP. Сценарій не настільки безпечний, як у порівнянні з іншими програмами. Все залежить від того, наскільки чутливим є додатковий вміст розробленого веб-сайту[40].

5. Форматування таблиць та графіки. Змінення форматування таблиць або графіки складніше, ніж в інших програмах, таких як Joomla.

6. SQL запити. Запити можуть бути складними через те, що всі додаткові користувальницькі поля, що створюються в WordPress вимагають додаткового синтаксису.

WordPress – це дуже хороший сценарій для запуску блогу або створення простого веб-сайту. Це один з простих у використанні рішень, доступних на ринку. Він також пропонує відмінний набір для початківців [28]. Для більш просунутих користувачів з більшими потребами, буде рекомендовано встановити безліч плагінів, щоб отримати більшу функціональність.

Drupal є однією з найбільш повних систем CMS, доступних на ринку. Сценарій має дуже специфічне середовище програмування, яке можна визначити як платформу програміста, ніж просто інструмент CMS [27].

Цей сценарій представляє собою варіант Open Source та є вибором багатьох програмістів, які раніше використовували прості інструменти, які мають не достатньо функцій, наприклад, WordPress.

Переваги Drupal [26]:

1. Майже всі основні функції доступні. Стандарт Drupal, як стандарт, надає такі функції, як реєстрація користувачів і керування профілем, RSS-канали, керування меню, система адміністрування та інструмент налаштування сторінок. Ці функції дозволяють розробнику створювати прості сторінки, один або декілька блогів користувача, форум, соціальну мережу тощо.

2. Управління дизайном веб-сайтів. Сценарій пропонує основні можливості редагування елементів веб-сайту. Завдяки використанню шаблонів та тем, немає необхідності розпочати все з нуля. Визначені конфігурації функцій веб-сайту полегшують створення простих та складних макетів сторінок.

3. Різні типи вмісту. Drupal славиться тим, що дозволяє розробнику створювати та керувати багатьма типами вмісту, такими як відео, опитування, управління користувачами, текст, блог, подкасти, статистичні дані тощо.

4. Розширене адміністрування користувачів. Адміністратор може створювати нові облікові записи користувачів зі своїми правами. Користувачів можна розділити на групи та призначити ролі. Вони можуть керувати частиною розробленого сайту після надання відповідних дозволів.

5. Організація контенту веб-сайту. Одним з найважчих етапів багатьох CMS є організація вмісту для подальшого використання. Drupal дозволяє класифікувати вміст через URL-адреси, створювати власні типи списків. Ця структура дозволяє легко організувати, шукати та повторно використовувати вміст.

6. Сценарії. Drupal має кілька тисяч плагінів, доступних для розробки веб-сторінок. Оскільки Drupal є частиною Open Software, можна використовувати та створювати власні плагіни[40].

7. Технічна підтримка. На домашній сторінці Drupal є документація, великий дискусійний форум, чат, список розсилки тощо. Тут розробники знайдуть багато інформації та допомогу щодо використання та зміни сценарію.

Недоліки Drupal [27].

1. Встановлення та модифікація. Цей сценарій не надто «зручний для користувача» і вимагає більш просунутих знань для його встановлення або зміни.



Quick.Cms, WordPress або навіть Joomla простіше використовувати, хоча вони не пропонують таку багату функціональність, як Drupal[37].

2. Сумісність. Drupal є абсолютно новим продуктом, тому він має досить багато нових рішень. Якщо розробник орієнтується на використання старих систем, йому буде потрібно час, щоб звикнути до цього сценарію.

3. Продуктивність. Зважаючи на масштабованість та продуктивність, Drupal відстає від скриптів, таких як Quick.Cms або WordPress. Якщо на трозробленому сайті передбачається багато трафіку, сценарій буде генерувати велику кількість завантажень для сервера. Це пов'язано з широким спектром можливостей, які надає Drupal [26]. Тим не менш, існують плагіни для створення веб-сайту до кеш-пам'яті сервера, що знижує його навантаження.

Drupal – це потужна система CMS, яка надає багато функцій, які в інших скриптах вимагають встановлення кількох плагінів. Але будь-яке втручання у код скрипта вимагає більш просунутих знань мов веб програмування – PHP, HTML та CSS [16]. Існує технічна підтримка, яка дуже широко представлена на англійській мові. Використовуючи цей розширений сценарій, слід пам'ятати, що при збільшенні трафіку на веб-сайті адміністратор сервера може вимкнути розроблений веб-сайт через надмірну завантаженість[40].

Joomla! – це скрипт, який є чимось між простим Wordpress, а також всеосяжним і вдосконаленим Drupal. Більш просунута, ніж WordPress, але менш складна, ніж Drupal [26].

Сценарій типу Open Source – це найпоширеніший вибір багатьох програмістів для розробки розширених веб-сайтів.

Переваги сценарію [27].

1. Дуже проста установка. Joomla! його можна легко встановити. Після завантаження для запуску сценарію на сервері потрібно лише кілька хвилин.

2. Сценарії. Цьому скрипту доступно кілька тисяч безкоштовних плагінів. Можна зазначити, що у WordPress їх більше, але щоб наблизити його до функціональності Joomla! розробникам потрібно виконати встановлення десятка плагінів.

3. Технічна підтримка. Joomla! має величезний вибір інструментів програмування, посібників, доступних для користувачів. Має дуже широкий дискусійний форум англійською мовою. Існує також велика спільнота користувачів на рідній мові.

4. Управління навігацією. Сценарій має комплексну навігаційну систему, яка може успішно підтримувати кілька ієрархій. Це дозволяє легко керувати веб-сайтом, якщо він містить сотні різних підсторінок.

5. Прості URL-адреси. Посилання, створені сценарієм, дуже прості, і вони впливають на рейтинги пошукових систем.

6. Просте оновлення. Після створення сайту сценарій потрібно оновити до нової версії. Розробник може це зробити в браузері. Потрібна лише обробка тексту.

7. Розширені інструменти адміністрування. Панель адміністрування надає багато функцій, які спочатку вражають своїм різноманіттям. Проте з часом більшість з них можуть бути освоєні, і завдяки цьому розробник може повністю використовувати сценарій.

Недоліки сценарію [28].

1. Обмежені можливості налаштування. Хоча Joomla! має безліч модулів і шаблонів, але для просунутих користувачів може не вистачати функцій.

2. Ресурси та ефективність. Для сценарію потрібен сервер з досить хорошими параметрами, оскільки він займає багато ресурсів в порівнянні з іншими системами CMS. Тому, чим більш розвинений і функціональний сайт, тим повільніше він буде працювати.

3. Витрати. Є плагіни та шаблони для Joomla!, які є платними, на відміну від WordPress, де всі плагіни та шаблони є безкоштовними.

4. Сумісність модулів. Іноді виникає складність користувацької сумісності між деякими модулями та плагінами. Це може призвести до того, що розробники не зможуть отримати функціональність без великої кількості змін у коді PHP.

5. Велика кількість можливостей. Багато користувачів, особливо тих, хто є початківцями, можуть бути вражені великою кількістю можливостей та функціями

CMS. Тому, якщо сайт повинен бути простим і користувач не має великої кількості знань, варто використовувати Quick.Cms або WordPress.

Якщо розробник не є новачком CMS та є необхідність в функціях більше, ніж, WordPress, але немає необхідності використання розширеного Drupal, то Joomla! є найбільш оптимальним рішенням [27]. Сценарій безкоштовний, має безліч плагінів, модулів та шаблонів, а також має дуже гарну технічну підтримку та дуже популярний в Україні.

ТуроЗ – це дуже потужна та розгалужена система CMS з чудовими можливостями. Цей сценарій, розроблений як Open Source, - це вибір більш просунутих програмістів, які шукають комбінації, який підтримує безліч функціональних можливостей за рахунок великої кількості інвестицій, щоб дізнатися про це [28].

#### Переваги сценарію

1. Модульність. Мета, для якої була створена ТуроЗ – це модульність та універсальність. Завдяки його розширенню він може добре працювати як сценарій для веб-сайту компанії та для обслуговування великого інформаційного порталу.

2. Розширення. Система пропонує кілька тисяч розширень, які встановлюються досить швидко і легко. Розробник зможе розширити функціональність за допомогою додаткових фотогалерей, дискусійного форуму, інтернет-магазину тощо.

3. Гнучка система адміністрування. Багато систем дозволяють отримувати доступ лише одному адміністратору. Однак у ТуроЗ користувачі можуть отримувати різні права, щоб вони могли бути звичайними користувачами, які використовують ресурси сайту, і можуть бути адміністраторами, які мають доступ до керування цими ресурсами. Крім того, користувачі можуть бути розділені, наприклад, на редакційні команди.

4. Типи вмісту. ТуроЗ має багато типів вмісту, які розробник може визначити. Завдяки цьому можна контролювати зовнішній вигляд веб-сайту, додавши фотографії, форми, таблиці, мультимедіа або звичайний текст.

5. Технічна підтримка. На домашній сторінці ТуроЗ існує велика документація та дискусійний форум. Там можна знайти багато інформації та допомогу щодо роботи та модифікації сценарію.

6. Скасування змін. Система має великий модуль зворотного зв'язку. У випадку великого порталу, над яким працюють багато розробників, це дуже бажаний варіант. Є можливість відновити попередню версію вмісту сайту практично без обмежень.

7. Зміна структури сторінки. Однією з найважливіших переваг є можливість організації структури обслуговування без поділу «зверху вниз» на блоки, розділи або статті.

8. Внутрішня мова TurboScript. Сценарій має внутрішню мову типу TurboScript, яка дозволяє створити багато елементів і розмістити в ньому динамічний вміст [16].

#### Недоліки сценарію

1. Встановлення та модифікація. ТуроЗ непросто встановити та налаштувати. Потрібен час, щоб дізнатися, як його використовувати, і розробнику обов'язково потрібно буде більше часу присвятити ніж іншим скриптам, таким як WordPress або Joomla. Тому не рекомендується для початківців.

2. Ресурси та продуктивність сервера. На жаль, модульність і розширюваність часто йдуть рука об руку з більшими вимогами параметрів сервера. Звичайно, якщо сайт не надто великий і не буде відвідуваний тисячами користувачів, на початку не повинно бути серйозних проблем.

3. Технічна підтримка. Сценарій має добре описану документацію та широкий дискусійний форум, але англійською мовою. ТуроЗ в Україні ще не настільки популярний, і розробнику знадобиться знання англійської мови, щоб мати знання про особливості платформи [26].

ТуроЗ, хоча є універсальним та придатним для створення простих веб сторінки, а також розширених порталів з різними типами вмісту, для програмування потрібен багато часу[40]. Модульність та розширюваність сценарію є його основними перевагами, але це досягається за рахунок продуктивності та вимог до

серверних ресурсів [27]. Тому при розробці достатньо простих веб сайтів використання ТуроЗ є недоречним.

Визначивши декілька платформ, проаналізувавши їх основні функції та властивості виконаємо порівняння цих платформ за допомогою деяких критеріїв прийняття рішень.

Метод аналізу ієрархії. Цей метод представляє собою механізм декомпозиції проблеми на все більш прості складові частини і подальшій обробці послідовності суджень особи, яка приймає рішення (ОПР), по парним порівнянь [23].

Для порівняння зупинимося на наступних платформах:

А - Розробка веб системи методом власного написання коду;

Б - Drupal;

В -WordPress.

Платформи було обрано, тому що WordPress є найбільш простою, а вибір Drupal обумовлено тим, що він безкоштовний на відміну від Joomla.

Визначимо критерії по яким будемо порівнювати засоби розробки веб системи управління:

- 1) надійність платформи;
- 2) взаємодія з системами управління базами даних;
- 3) важкість розробки;
- 4) функціональність продукту;
- 5) можливість внесення змін;
- 6) безпека даних;
- 7) складність розгортання платформи;
- 8) швидкість роботи.

Виконаємо побудову матриці парних порівнянь (таблиця 2.1).

Таблиця 2.1. Матриця парних порівнянь

Загальне визначення платформи	Надійність	Взаємодія	Швидкість	Функціональність	Внесення змін	Безпека	Складність розгортання	Важкість розробки
Надійність	1	5	3	7	6	6	1/3	1/4
Взаємодія	1/5	1	1/3	5	3	3	1/5	1/7
Швидкість	1/3	3	1	6	3	4	6	1/5

функціональність	1/7	1/5	1/6	1	1/3	1/4	1/7	1/8
Внесення змін	1/6	1/3	1/3	3	1	1/2	1/5	1/6
безпека	1/6	1/3	1/4	4	2	1	1/5	1/6
Складність розгортання	3	5	1/6	7	5	5	1	1/2
Важкість розробки	4	7	5	8	6	6	2	1

Клітини матриці заповнюються суб'єктивними оцінками, що визначають важливість того чи іншого критерію, шкалу взято від 1 до 9, де 1 не має значення, 9 найбільш важливо. У той же час повина зберігатися симетрія в матриці – якщо в комірці 1-2 міститься значення 5, то в комірці 2-1 повинно міститися 1/5 [23].

Виконаємо розрахунки:

1) знаходимо значення першого рядка  $\sqrt[8]{\phantom{x}}$

$$\sqrt[8]{1 \times 5 \times 3 \times 7 \times 6 \times 6 \times 1/3 \times 1/4} = 2,052;$$

аналогічно обчислюються власні вектора інших рядків.

2) Проводимо нормалізацію отриманих чисел (складаємо значення рядків)

$$r = 2,052 + 0,74 + 1,75 + 0,23 + 0,42 + 0,5 + 1,57 + 4,11 = 11,372.$$

3) Визначаємо вектор пріоритетів для першого рядка

$$2,052 / 11,372 = 0,18.$$

Аналогічно обчислюються значення інших рядків.

4) Визначаємо найбільше значення матриці суджень для рівня 2:

$$\lambda_{\max} = (9,01 \times 0,18) + (21,86 \times 0,065) + (10,25 \times 0,153) + (41 \times 0,02) + (23,3 \times 0,037) + (25,75 \times 0,044) + (10,08 \times 0,138) + (2,25 \times 0,362) = 9,73;$$

## 5) Індекс узгодженості (IC) дорівнює

$$IC = (\lambda_{\max} - n) / (n - 1) = (9,73 - 8) / 7 = 0,247;$$

Відношення узгодженості (OC) отримаємо шляхом ділення IC на число, відповідне випадкової узгодженості матриці того ж порядку.

Для матриці  $8 \times 8$  випадкова узгодженість дорівнює 1,41.

$$OC = IC / 1,41 = 0,247 / 1,41 = 0,175.$$

Таким чином:

$$\lambda_{\max} = 9,73$$

$$IC = 0,247$$

$$OC = 0,175$$

Розглянемо вектори пріоритетів. Найбільше значення має вектор «Важкість розробки». Ясно, що саме розробка веб системи управління, її програмна реалізація представляє найбільш дієву зацікавленість для розробника.

Виконаємо обчислення для рівня 3 пріоритетів, найбільшого власного значення суджень, індексу узгодженості відносин узгодженості для всіх восьми матриць суджень розмірністю  $3 \times 3$  [19]. Для цього виконаємо побудову матриці суджень (таб. 2.2).

Таблиця 2.2. Матриця суджень для рівня 3

Надійність	<i>A</i>	<i>B</i>	<i>V</i>	Взаємодія	<i>A</i>	<i>B</i>	<i>V</i>
<i>A</i>	1	6	8	<i>A</i>	1	7	1/5
<i>B</i>	1/6	1	4	<i>B</i>	1/7	1	1/8
<i>V</i>	1/8	1/4	1	<i>V</i>	5	8	1
Швидкість	<i>A</i>	<i>B</i>	<i>V</i>	Функціональність	<i>A</i>	<i>B</i>	<i>V</i>

Таблиця 2.2 (Продовження)

<i>A</i>	1 8 6	<i>A</i>	1 1 1
<i>B</i>	1/8 1 1/4	<i>B</i>	1 1 1
<i>B</i>	1/6 4 1	<i>B</i>	1 1 1
Внесення змін	<i>A B B</i>	Безпека	<i>A B B</i>
<i>A</i>	1 5 4	<i>A</i>	1 8 6
<i>B</i>	1/5 1 1/3	<i>B</i>	1/8 1 1/5
<i>B</i>	1/4 3 1	<i>B</i>	1/6 5 1
Складність розгортання	<i>A B B</i>	Важкість розробки	<i>A B B</i>
<i>A</i>	1 1/2 1/2	<i>A</i>	1 1/7 1/5
<i>B</i>	2 1 1	<i>B</i>	7 1 3
<i>B</i>	2 1 1	<i>B</i>	5 1/3 1

В таблиці 2.3 представлені обчислення результатів суджень.

Таблиця 2.3. Обчислення результатів суджень

Надійність	<i>A B B</i>	Вектор пріоритетів	Внесення змін	<i>A B B</i>	Вектор пріоритетів
<i>A</i>	1 6 8	0,754	<i>A</i>	1 5 4	0,674
<i>B</i>	1/6 1 4	0,181	<i>B</i>	1/5 1 1/3	0,101
<i>B</i>	1/8 1/4 1	0,065	<i>B</i>	1/4 3 1	0,226
		$\lambda_{\max} = 3,136$ $IC = 0,068$ $OC = 0,117$			$\lambda_{\max} = 3,06$ $IC = 0,043$ $OC = 0,074$
Взаємодія	<i>A B B</i>	Вектор пріоритетів	Безпека	<i>A B B</i>	Вектор пріоритетів
<i>A</i>	1 7 1/5	0,233	<i>A</i>	1 8 6	0,747
<i>B</i>	1/7 1 1/8	0,005	<i>B</i>	1/8 1 1/5	0,06
<i>B</i>	5 8 1	0,713	<i>B</i>	1/6 5 1	0,193
		$\lambda_{\max} = 3,247$ $IC = 0,124$ $OC = 0,213$			$\lambda_{\max} = 3,197$ $IC = 0,099$ $OC = 0,17$
Швидкість	<i>A B B</i>	Вектор пріоритетів	Складність розгортання	<i>A B B</i>	Вектор пріоритетів
<i>A</i>	1 8 6	0,745	<i>A</i>	1 1/2 1/2	0,2
<i>B</i>	1/8 1 1/4	0,065	<i>B</i>	2 1 1	0,4
<i>B</i>	1/6 4 1	0,181	<i>B</i>	2 1 1	0,4
		$\lambda_{\max} = 3,13$ $IC = 0,068$ $OC = 0,117$			$\lambda_{\max} = 3$ $IC = 0$ $OC = 0$



Таблиця 2.3 (Продовження)

Функціональність	<i>A B B</i>	Вектор пріоритетів	Важкість розробки	<i>A B B</i>	Вектор пріоритетів
<i>A</i>	1 1 1	0,333	<i>A</i>	1 1/7 1/5	0,072
<i>B</i>	1 1 1	0,333	<i>B</i>	7 1 3	0,65
<i>B</i>	1 1 1	0,333	<i>B</i>	5 1/3 1	0,278
		$\lambda_{\max} = 3$ $IC = 0$ $OC = 0$			$\lambda_{\max} = 3,065$ $IC = 0,032$ $OC = 0,056$

Стовпці матриці заповнюються значеннями векторів пріоритетів, які беруться з матриці парних порівнянь для рівня 3.

Уявімо матрицю глобальних пріоритетів у вигляді таблиці 2.4.

Таблиця 2.4. Вибір платформи: матриця глобальних пріоритетів

	Вектори пріоритетів								Узагальнення або глобальні пріоритети
	1 (0,18)	2 (0,065)	3 (0,153)	4 (0,02)	5 (0,037)	6 (0,044)	7 (0,138)	8 (0,362)	
<i>A</i>	0,754	0,233	0,745	0,333	0,674	0,747	0,2	0,072	0,37
<i>B</i>	0,181	0,055	0,065	0,333	0,101	0,06	0,4	0,065	0,341
<i>B</i>	0,065	0,713	0,181	0,333	0,226	0,193	0,4	0,278	0,263

Глобальний пріоритет обчислюється шляхом множення векторів пріоритетів другого рівня на вектори пріоритетів третього рівня, потім результати складаються уздовж кожного рядка. Наприклад, для варіанту розробки *A* маємо:

$$(0,754 \times 0,18) + (0,233 \times 0,065) + (0,745 \times 0,153) + (0,333 \times 0,02) + (0,647 \times 0,037) + (0,747 \times 0,044) + (0,2 \times 0,138) + (0,072 \times 0,362) = 0,37$$

Висновок: варіант побудови платформи *A* (розробка методом написання власного програмного коду), який був найменш бажаний з точки зору важкості розробки (критерій з найвищим пріоритетом), всупереч очікуванню виявився переможцем.

## Висновки по розділу 2

В рамках цього розділу були вирішені наступні задачі:

- визначено користувачів веб системи управління науковою та педагогічною роботою кафедри, побудовані діаграми варіантів використання;
- побудовано алгоритмічне забезпечення задачі, представлена робота основних підсистем;
- проведено аналіз існуючих платформ на яких відбувається побудова веб систем управління;
- проведено вибір середі розробки за допомогою методу аналізу ієрархій, на основі якого визначено, що найбільш оптимальним рішенням для розробки веб системи управління науковими та педагогічними аспектами роботи кафедри є розробка за допомогою написання власного коду.

Таким чином, було виконана підготовка для фізичної розробки веб системи управління.

### **3. ОПИС ПРОГРАМНОЇ РОЗРОБКИ ВЕБ СИСТЕМИ УПРАВЛІННЯ НАУКОВОЮ ТА ПЕДАГОГІЧНОЮ РОБОТОЮ КАФЕДРИ З МОДУЛЕМ ФОРМУВАННЯ ПОВІДОМЛЕНЬ**

#### **3.1 Побудова діаграми класів**

Діаграма класів базується на понятті «Об'єкт». Об'єктом є структура даних, яка становить у реалізації комп'ютера зіставлення істоти, яка має чітко визначені межі і властивості. Концепція об'єкта тісно пов'язана з наступними поняттями [2]:

1. Ідентичність об'єкта, який відрізняє його від інших об'єктів і не залежить від значення його атрибутів, зв'язків з іншими об'єктами, розташування (відображуваного об'єктом), що знаходиться в реальному світі, і місця розташування об'єкта в адресному просторі комп'ютера[33].

2. Стан об'єкта, яке визначається поточними значеннями його атрибутів і посиланнями на інші об'єкти. Стан об'єкта може змінюватися з часом; однак це не тягне за собою зміни його особистості.

3. Поведінка, призначена об'єкту, тобто набір операцій, які можуть виконуватися на ньому.

4. Тип, призначений об'єкту, тобто лінгвістичне вираження, яке за допомогою вказівки атрибутів визначає його структуру і обмежує контекст, в якому можна посилатися на об'єкт.

Всі атрибути, асоціації, операції і т.д. об'єкта називаються його властивостями [18].

Об'єкт, який є відтворенням певного елемента з реального світу, повинен зберігати всю інформацію, що стосується цього елементу[45]. Об'єкт передає цю інформацію між іншими об'єктами через механізм атрибутів. Атрибути об'єкта можуть бути самими об'єктами, в цьому випадку званими подоб'єктами; об'єкт, що містить атрибути підоб'єкту, називається складовим об'єктом [3].

Клас, як об'єкт, відноситься до основних понять об'єктно-орієнтованого підходу.

Клас – це іменований набір об'єктів зі схожими властивостями: семантика, атрибути, поведінку, стосунки з іншими об'єктами і т.д. Згідно з цим визначенням, клас не є набором об'єктів, але використовується для оголошення об'єктів [12].

Найбільш важливими властивостями класу, званими інваріантами класу, є:

- ім'я або лінгвістичний ідентифікатор класу об'єкта; поля атрибутів;
- тип, тобто структура (конструкція) об'єкта, зазначеного з використанням механізму атрибута;
- методи або реалізації операцій, які можуть виконуватися на об'єкті [3].

Інші типи інваріантів включають, серед іншого [2,46]:

- обробки подій і винятків, які можуть виникати при виконанні операцій на сайті;
- список експорту, тобто список властивостей об'єкта, доступних ззовні;
- обмеження, на які може піддаватися об'єкт класу.

У мові UML, в діаграмі класів, визначення класу представлено як прямокутник, що складається з трьох полів [15]:

- поля класу імен;
- поля атрибутів;
- поля методу.

Існує три основних види доступності, позначені наступним чином:

- +: публічний елемент - це видно з-поза класу;
- : приватний елемент – його видно лише всередині класу;
- #: захищений елемент - видно в класі та його підкласах [14].

Кожен елемент специфікації класу, крім назви, є обов'язковим; зокрема, дуже часто на стадії аналізу або, принаймні, на початку, доступність та типи атрибутів не визначаються [3,42].

На рисунку 3.1 представлено діаграму класів веб системи управління науковими та педагогічними аспектами роботи кафедри.

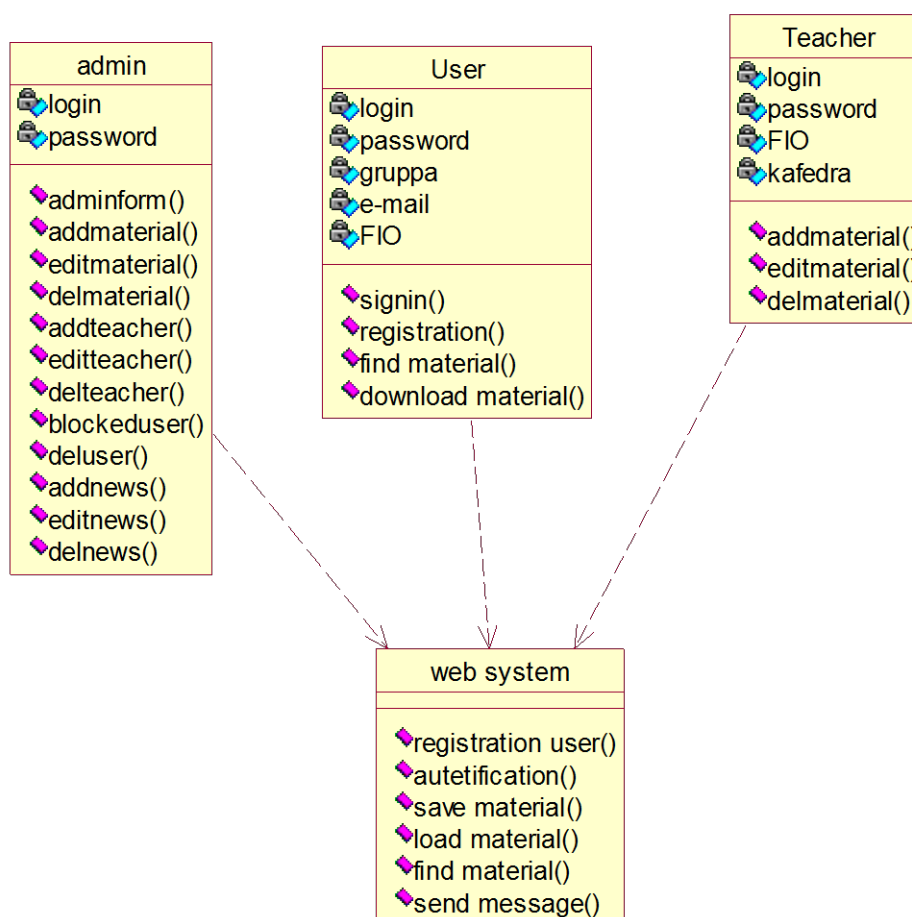


Рисунок 3.1 – Діаграма класів веб системи

Представимо призначення полів представлених класів. В таблиці 3.1 наведено значення полів для класу «Admin».

Таблиця 3.1. Поля класу «Admin»

Назва поля	Призначення
Атрибути	
login	Логін адміністратора
password	Пароль входу адміністратора
Продовження таблиці 3.1	
Методи (операції)	
adminform	Користування адмін. Формою веб системи
addmaterial	Додавання матеріалів
editmaterial	Редагування матеріалів
delmaterial	Видалення матеріалів
addteacher	Додавання викладача

Таблиця 3.1 (Продовження)

Editteacher	Редагування даних по викладачу
delteacher	Видалення викладача
blockeduser	Блокування облікового запису студента
deluser	Видалення студенту
addnews	Додавання новин
editnews	Редагування новин
delnews	Видалення новин

В таблиці 3.2 наведено значення полів для класу «User».

Таблиця 3.2. Поля класу «User»

Назва поля	Призначення
Атрибути	
login	Логін студента
password	Пароль входу студента
gruppa	Група студента
e-mail	Адрес електронної поштової скриньки
FIO	Прізвище, ім'я по батькові студента
Операції	
signin	Вхід в веб систему
registration	Реєстрація в системі
find material	Пошук матеріалів
download material	Завантаження матеріалів

В таблиці 3.3 наведено значення полів для класу «Teacher».

Таблиця 3.3. Поля класу «Teacher»

Назва поля	Призначення
Атрибути	
login	Логін студента
password	Пароль входу студента
kafedra	Назва кафедри
FIO	Прізвище, ім'я по батькові викладача
Операції	
addmaterial	Додавання матеріалів
editmaterial	Редагування матеріалів
Продовження таблиці 3.3	
delmaterial	Видалення матеріалів

В таблиці 3.4 наведено значення полів для класу «Web system».

Таблиця 3.4. Поля класу «Web system»

Назва поля	Призначення
Атрибути	
Операції	
registration user	Реєстрація користувача
autentification	Аутентифікація користувача
save material	Зберегти матеріали
download material	Завантажити матеріали
find material	Знайти матеріали
send message	Надіслати повідомлення

### 3.2 Побудова діаграми виконання

UML визначає два типи діаграм виконання:

Компонентні діаграми – ілюструють структуру розробленого коду системи, вказуючи виконання елементів проекту (наприклад, класів або підсистем) за допомогою компонентів та їх інтерфейсів, а також вказуючи на взаємозв'язок між компонентами [3,44]. Метою ідентифікації компонентів є побудова систем достатньо високої якості, заповнення бажаних бізнес-потреб та швидкої побудови за рахунок складання готових деталей, які дозволяють швидше будувати систему ніж розробляти кожен елемент самотійно [14].

Діаграми впровадження призначенні для того, щоб показати конфігурацію системи часу виконання, тобто розподіл компонентів та об'єктів на вузлах. Вузли моделюють обчислювальні ресурси часу виконання[34]. Ця конфігурація може бути як статичною, так і динамічною: компоненти та об'єкти можуть мігрувати між вузлами під час виконання [15,46].

Компонент є добре ізольованим від контексту блоку реалізації, узгоджений завдяки виконуваним функціям та чітко визначеному інтерфейсу. Правильно побудований компонент використовує інші компоненти лише через їх інтерфейси, що допомагає майбутнім модифікаціям [3]. Можна виділити кілька типів

компонентів: компоненти з точки зору одного проекту, компоненти з точки зору всього програмного забезпечення в даній організації та бізнес-компоненти.

На рисунку 3.2 представлена діаграма компонентів для веб системи управління.

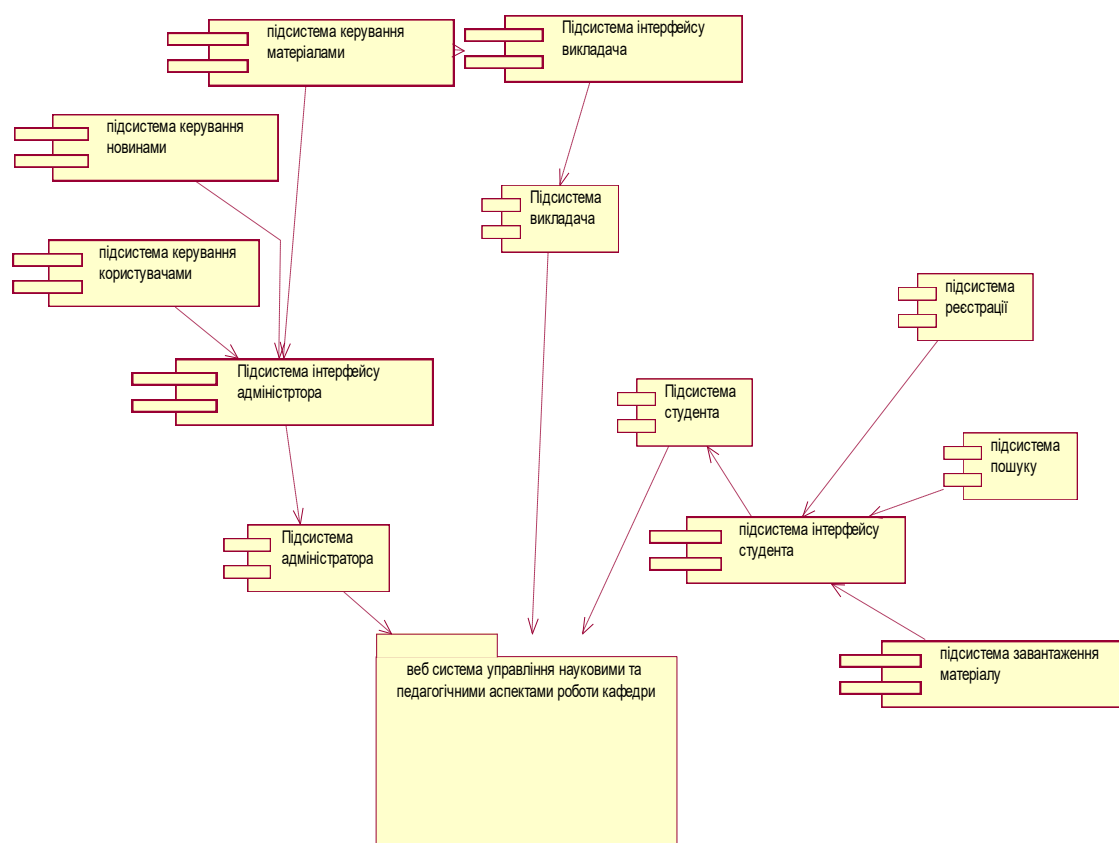


Рисунок 3.2 – Діаграма компонентів веб системи

Діаграми впровадження показують конфігурацію наступних елементів виконання [12,50]:

- апаратні компоненти, тобто фізичні одиниці з пам'яттю, а часто і обчислювальна потужність;
- компоненти програмного забезпечення (виконуваний код);
- об'єкти, пов'язані з компонентами.

Діаграма реалізації являє собою графік, де вершини, звані вузлами, з'єднуються лініями, які відображають комунікаційні з'єднання апаратних



компонентів [14]. На рисунку 3.3 представлено діаграму впровадження веб системи управління.

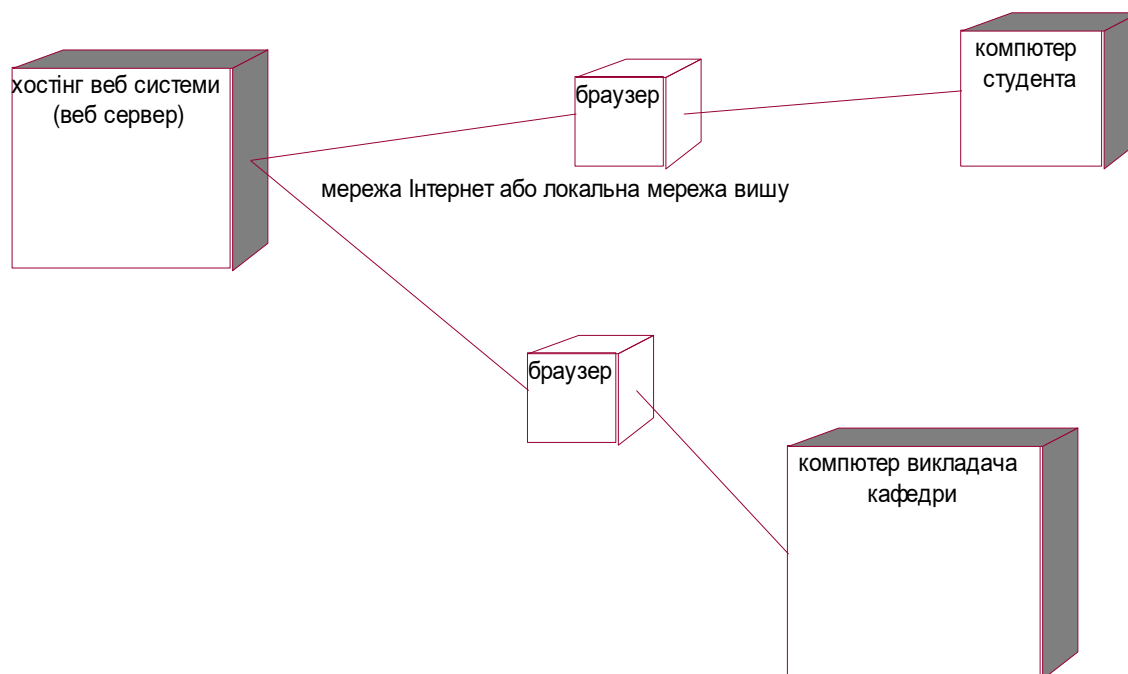


Рисунок 3.3 – Діаграма впровадження веб системи

### 3.3 Побудова логічної моделі бази даних веб системи

Побудова логічної схеми БД виконаємо в пакеті ERwin Data Modeler. CA ERwin Data Modeler (CA ERwin DM), інструмент проектування бази даних що підвищує рівень якості даних в транзакційних і сховищах даних [5,47].

В даному пакеті можна створювати і підтримувати графічні моделі, що представляють бази даних. CA ERwin DM забезпечує платформу, де вимоги до корпоративних даних і відповідні проекти баз даних можуть бути певними, керованими і реалізованими в безлічі баз даних [6].

Графічний користувацький інтерфейс на базі Windows поєднується з потужним інструменти прив'язки об'єктів (ER), до призначених для користувача редакторами для визначення фізичних об'єктів бази даних, з моделлю Explorer для

текстового представлення об'єктів моделі і з підтримкою провідних баз даних SQL [19].

Крім того, даний продукт може використовуватися для оптимізації розробки додатків процесу, дозволяючи різним групам здійснювати самостійну роботу[48]. Таким чином, різні групи можуть одночасно працювати з різними частинами моделі або різними типами моделей [11].

Щоб організувати базу даних для веб системи управління науковими та педагогічними аспектами роботи кафедри можна застосувати три способи:

- файлову організацію ІБ;
- інтегровану ІБ;
- змішану організацію ІБ [13].

При файлової організації база розміщується локально на комп'ютері. Інші користувачі здійснюють доступ до цього комп'ютера за допомогою стандартних методів ОС для мережевого обміну даними.

Змішана організація має на увазі розподілену базу даних, яка зберігається на декількох серверах і реплікує свої зміни згідно з визначеним розкладом. Таку структуру ІБ використовує в системах класу ERP, щоб працювати в одній інформаційній базі кількох віддалених офісах одночасно [5,44].

При інтегрованому способі організації інформаційна база представляється у вигляді сукупності взаємопов'язаних спільно збережених даних. При цьому забезпечується така мінімальна надмірність, при якій допустимо їх оптимальне використання в будь-яких додатках. Така організація забезпечує незалежність даних від програми, а щоб актуалізувати дані досить загального способу управління [11].

Для даної дипломної роботи найбільш доцільним є використання інтегрованої організації інформаційної БД. Так, це обумовлено в першу чергу значних щоденних зростанням бази - так, щодня в базу будуть додаватися 200-500 нових записів[41,49]. Також оптимальним рішенням буде застосування СУБД замість файлової організації бази даних.

Існує три моделі логічної структури бази даних (за способом встановлення зв'язків між даними): ієрархічна, мережева і реляційна [13,14].

Реляційна база даних – це база даних, в якій всі дані, доступні користувачеві, організовані у вигляді таблиць, а всі операції над даними зводяться до операцій над цими таблицями. Для представлення реляційних баз даних розроблена формальна теорія баз даних, теоретичну основу якої складає алгебра і математична логіка.

Реляційна модель орієнтована на організацію даних у вигляді двовимірних таблиць. Кожна реляційна таблиця являє собою двовимірний масив. Реляційна модель бази даних (relational database model) для зберігання і організації інформації використовує атрибути (стовпці) і записи (рядки). Реляційна модель бази даних в даний час є найбільш широко використовуваної моделлю [15].

Реляційна база даних складається з двовимірних таблиць, кожна таблиця містить унікальні рядки, стовпці і осередки. Кожна клітинка містить тільки одне значення даних, що представляє собою конкретне значення атрибута відповідного запису [15].

Ієрархічна модель бази даних (hierarchial data model) об'єднує пов'язані записи і поля в логічну деревоподібну структуру. Ця структура і взаємозв'язки між елементами даних, відрізняються від тих, які використовуються в реляційній базі даних. В ієрархічній базі даних батьківські елементи можуть мати дочірні елементи (один, кілька або жодного). Деревоподібна структура має гілки, кожна з них має безліч листів - полів даних [19,47]. У таких базах даних добре відомі, заздалегідь визначені шляхи доступу до даних, але вони не настільки гнучкі при створенні відносин між елементами даних, в порівнянні з реляційними базами даних. Ієрархічні бази даних доцільно використовувати для зберігання даних, що мають відношення «один-до-багатьох» [6].

Об'єктно-орієнтована база даних (object-oriented database) призначена для роботи з різними типами даних (зображення, аудіо, документи, відео). Система управління об'єктно-орієнтованими базами даних (ODBMS - object-oriented database management system) більш динамічна за своєю природою, ніж реляційна СУБД, оскільки вона створює об'єкти при необхідності, а дані і процедури (названі методами) при запиті об'єкта надаються разом з ним [11,50]. При роботі з реляційною базою даних, програма має використовувати свої власні процедури для

отримання даних з бази даних і їх обробки. Реляційна база даних не надає процедур, як це робить об'єктно-орієнтована база даних. Об'єктно-орієнтована база даних використовує класи для визначення атрибутів і процедур її об'єктів[39].

В якості моделі логічної структури бази даних була обрана саме реляційна. Така модель надає можливість швидкого сформування міжтабличних зв'язків, правильно будувати запитів до бази даних. Також така архітектура є більш простий і мінімізує час виконання запиту [23].

Логічна схема БД веб системи управління науковими та педагогічними аспектами роботи представлено на рисунку 3.4.

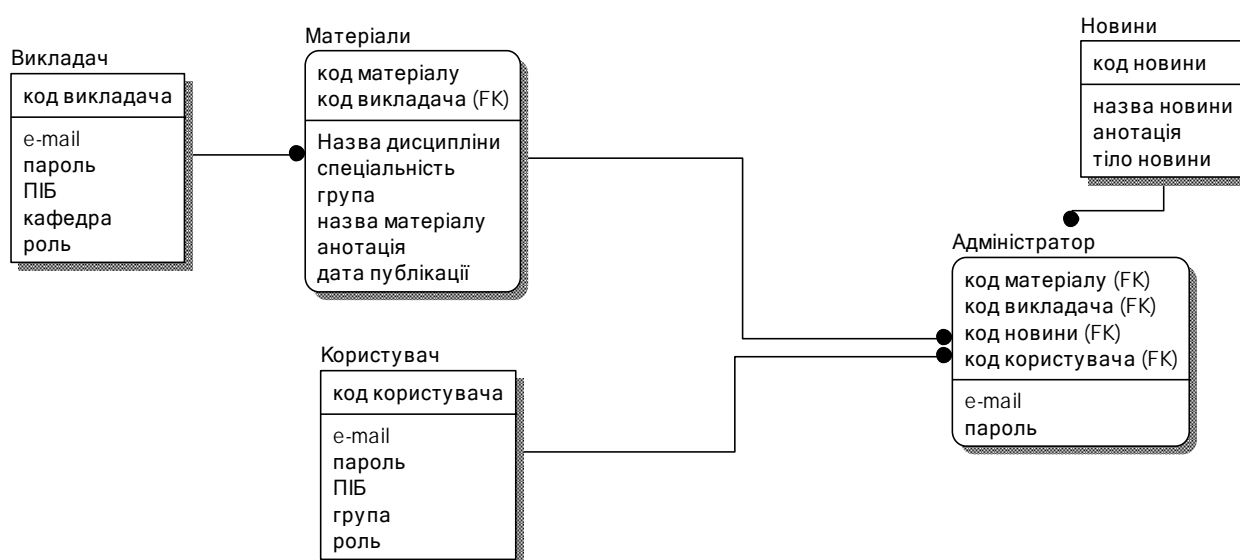


Рисунок 3.4 – Логічна схема бази даних веб системи управління

Представимо опис полів представлених таблиць бази даних. В таблиці 3.5 представлено опис таблиці «Викладач».

Таблиця 3.5. Таблиця «Викладач»

Назва поля	Тип даних
Код користувача	Лічильник
e-mail	Текстовий
Пароль	Текстовий
ПІБ	Текстовий
Кафедра	Текстовий
Роль	Текстовий

В таблиці 3.6 представлено опис таблиці «Користувач».

Таблиця 3.6. Таблиця «Користувач»

Назва поля	Тип даних
Код викладача	Лічильник
e-mail	Текстовий
Пароль	Текстовий
ПІБ	Текстовий
група	Текстовий
Роль	Текстовий

В таблиці 3.7 представлено опис таблиці «Матеріали».

Таблиця 3.7. Таблиця «Матеріали»

Назва поля	Тип даних
Код викладача	Лічильник
Код матеріалу	Лічильник
Назва дисципліни	Текстовий
Спеціальність	Текстовий
група	Текстовий
Назва матеріалу	Текстовий
Анотація	Текстовий
Дата публікації	Дата

В таблиці 3.8 представлено опис таблиці «Адміністратор».

Таблиця 3.8. Таблиця «Адміністратор»

Назва поля	Тип даних
Код викладача	Лічильник
Код матеріалу	Лічильник
Продовження таблиці 3.8	
Код новини	Лічильник
Код користувача	Лічильник
e-mail	Текстовий
Пароль	Текстовий

В таблиці 3.9 представлено опис таблиці «Новини».

Таблиця 3.9. Таблиця «Новини»

Назва поля	Тип даних
Код новини	Лічильник
Назва новини	Текстовий
Анотація	Текстовий
Тіло новини	Текстовий

### 3.4 Вибір середовища розробки

Після формування структури бази даних і її розробки, необхідно розробити програмний комплекс веб системи управління науковими та педагогічними аспектами роботи кафедри [1]. Як було описано раніше, система буде складатися з підсистем: клієнтської частини (частина викладача та студента), частини адміністратора і підсистеми управління сайтом, в зв'язку з цим програмна реалізація буде зводитися до розробки модулів для клієнта, і окремо для адміністратора, раніше були описані елементи управління веб системи, при розробці підсистема управління буде об'єднана з частиною адміністратора [22].

Для розробки добре функціонуючого веб-додатки - багатьма фахівцями вважається, що кращою комбінацією інструментів для створення веб додатки буде комбінація PHP і MySQL [16].

PHP – це швидка і багатофункціональна мова сценаріїв з відкритим вихідним кодом, який використовується для розробки веб-додатків або додатків Інтернет / Інтранет [17].

MySQL – це потужний сервер баз даних з відкритим вихідним кодом, побудований на основі системи керування базами даних (RDBMS) і здатний обробляти велику паралельне з'єднання з базою даних [31].

У поєднанні один з одним PHP і MySQL можуть створювати дуже потужні і масштабовані додатки Web / Internet / Intranet [24].

PHP і MySQL – це Open Source, що означає, що вони є безкоштовними інструментами розробки, і існує велика спільнота програмістів, які вносять свій внесок в удосконалення та постійно додають до нього функції [18]. Засоби розробки

та сервери баз даних, що вимагають витрат на ліцензування, мають обмежені ресурси програмування в порівнянні з інструментами розробки з відкритим вихідним кодом, які мають швидке зростання спеціалізованого та добре обізнаного співтовариства, яке поширюється по всьому світу [25].

PHP і MySQL є кращими інструментами розробки. При правильній розробці програми можуть бути створені з простою практичністю, складною функціональністю, швидкістю, потужністю і масштабістю [20].

PHP і MySQL побудовані так, що будь-який, хто має мінімальну кількість знань в області програмування, може об'єднати веб-додаток на основі браузера [21].

Для перевірки роботи програми за допомогою мови PHP та СУБД MySQL необхідно використовувати веб сервер, наприклад Денвер. Даний веб сервер має великий набір інструментів, що допомагають в розробці різних веб додатків, наприклад додаток RHPMyAdmin допомагає при розробці бази даних веб системи (див. рис.3.5) [25]. В цьому додатку доступні наступні операції над таблицями бази даних:

- визначати структуру БД, формуючи таблиці;
- формувати sql запити;
- шукати необхідну інформацію в полях таблиці;
- експортувати або імпортувати дані;
- об'єднувати таблиці або бази даних;
- видаляти таблиці.

Написання коду PHP може відбуватися як в середовищах, таких як PHP Editor, так і в звичайном блокноті [24]. На рисунку 3.6 представлено приклад розробки PHP кода для веб системи управління.

В результаті було розроблено веб систему управління. У комплект поставки програмного продукту входять:

- 1) dipl.loc - набір динамічних сторінок;
- 2) dipl - sql база, яка містить інформацію про зміну трафіку, користувачів і груп системи;
- 3) model.php - файл допомоги;

4) test - каталог з тестовими прикладами.

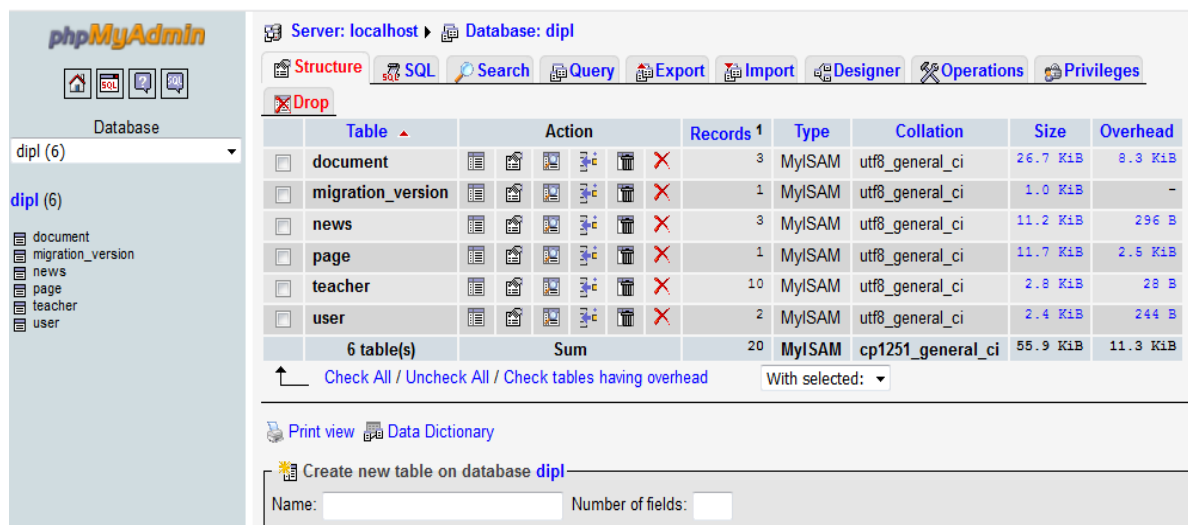


Рисунок 3.5 – Додаток RHPMyAdmin

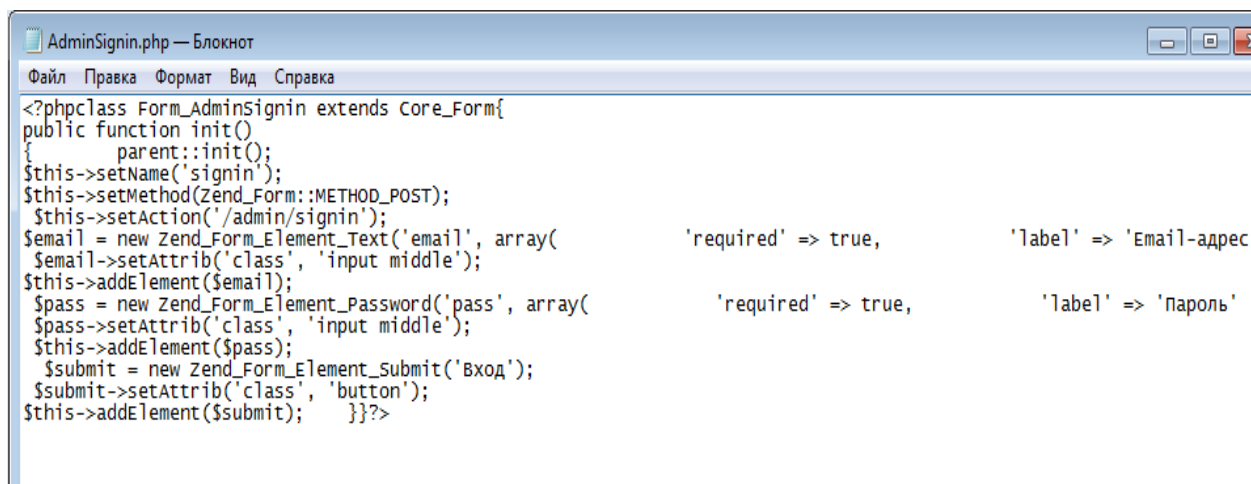


Рисунок 3.6 – Написання PHP коду у середовищі блокнота

Для запуску системи управління контентом навчального матеріалу необхідно встановити на комп'ютер веб-сервер Apache, який відрізняється надійністю і гнучкістю конфігурації. Він дозволить підключити зовнішні модулі для надання даних, використовує СУБД для аутентифікації користувачів, модифікує повідомлення про помилки. Після цього необхідно переписати файли з папки `dipl.dev` в локальну папку веб-сервера, а папку `dipl` в каталог баз sql. Після цього програмний комплекс готовий до роботи.



### **Висновки по розділу 3**

В рамках цього розділу були виконані наступні задачі:

- виконано побудову діаграми класів;
- розроблено діаграму компонентів та розгортання;
- побудовано логічну схему БД веб системи;
- визначене середовище розробки веб системи управління науковою та педагогічною роботою кафедри;
- побудовано карту веб системи.

## **4. МЕТОДИКА РОБОТИ КОРИСТУВАЧА З ПРОГРАМНОЮ СИСТЕМОЮ**

Розроблений програмний продукт призначений для використання користувачами як на комп'ютерах, так і на мобільних пристроях. Для перегляду результатів програмної реалізації необхідний лише пристрій (комп'ютер чи смартфон) з вбудованим браузером.

На рисунку 4.1 представлена головна сторінка розробленої веб системи управління науковими та педагогічними аспектами роботи кафедри.

Рисунок 4.1 – Головна сторінка веб системи

Для того щоб увійти в адміністраторську панель необхідно перейти на вкладку «Вхід» (див. рис.4.2).

Рисунок 4.2 – Сторінка входу користувачів

Для входу до панелі адміністратора необхідно ввести в адресі e-mail та пароль для авторизації. На рисунку 4.3 представлено панель адміністратора під час редагування головної сторінки веб системи.

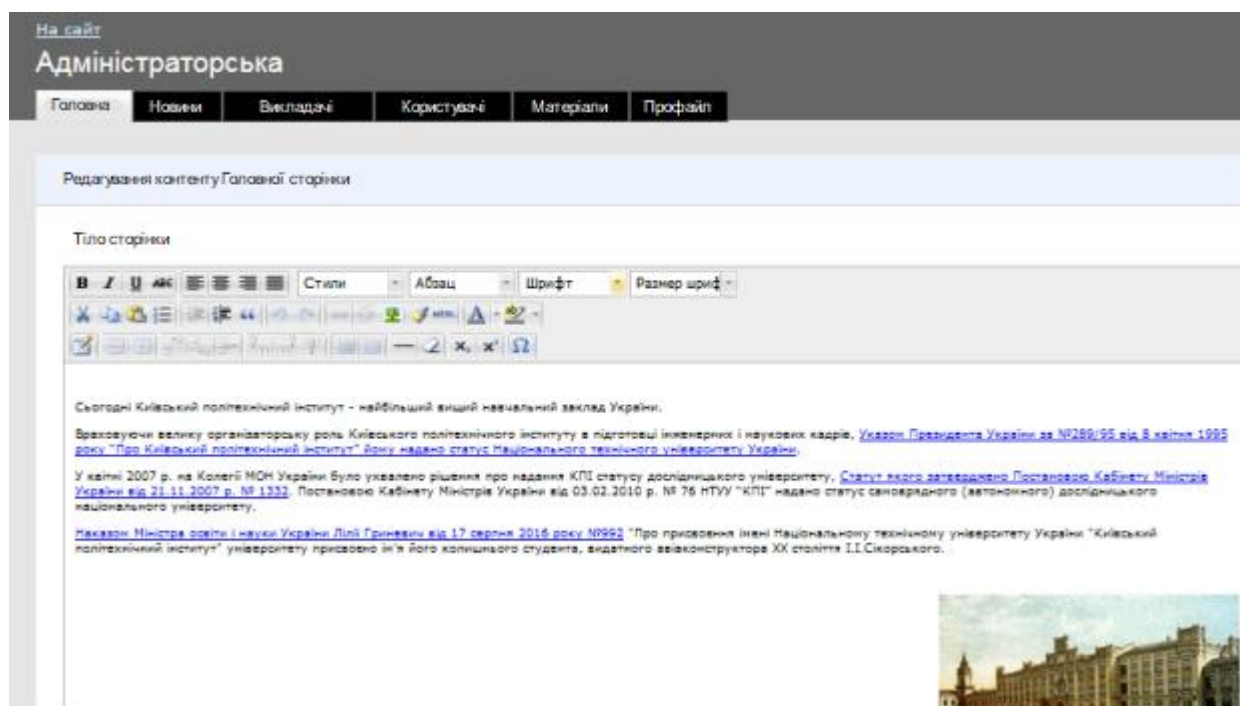


Рисунок 4.3 – Адміністраторська панель веб системи

Як видно з представленого рисунку адміністратор має наступні пункти меню:

- головна;
- новини;
- викладачі;
- користувачі;

- матеріали;
- профайл.

При додаванні матеріалу на головну сторінку адміністратор може скористуватися оболонкою, яка за своєю функцією схоже на панель пакета MS Word, ця оболонка дозволяє змінювати стилі тексту, формувати абзац, додавати списки, змінювати шрифт, додавати зображення.

При додаванні новин адміністратор повинен вказати назву новини, коротку анотацію та не посередньо текст новини (див. рис. 4.4).

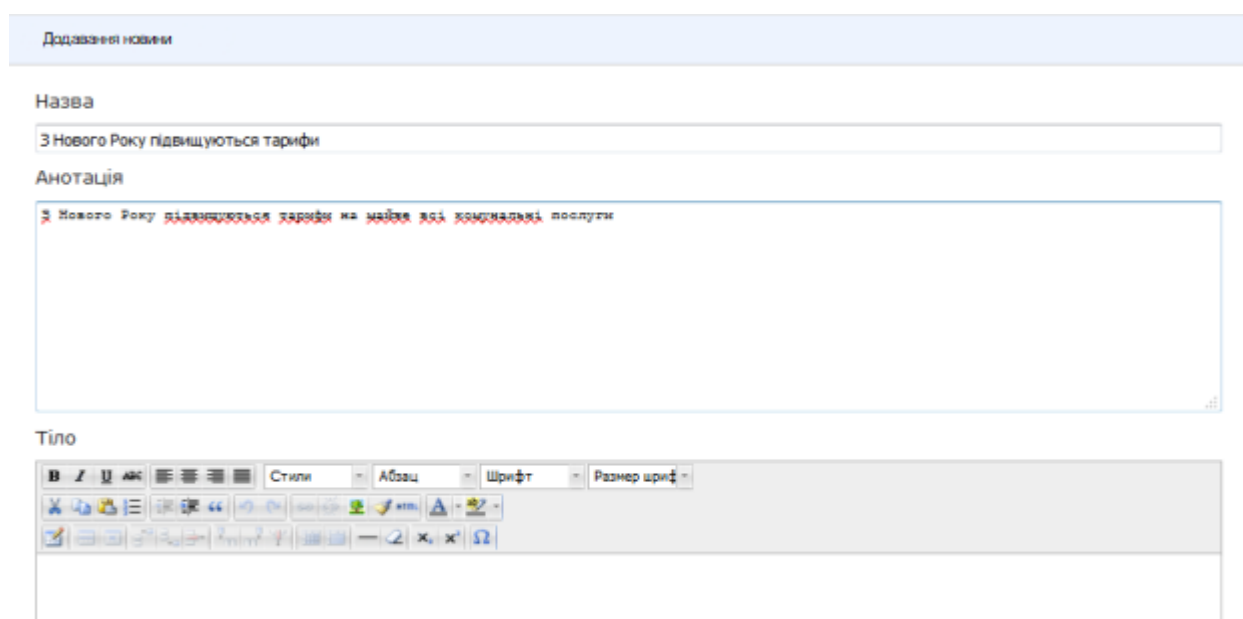


Рисунок 4.4 – Додавання новини

На рисунку 4.5 представлена панель управління користувачами веб системи. Користувачі автоматично додаються після проходження реєстрації.

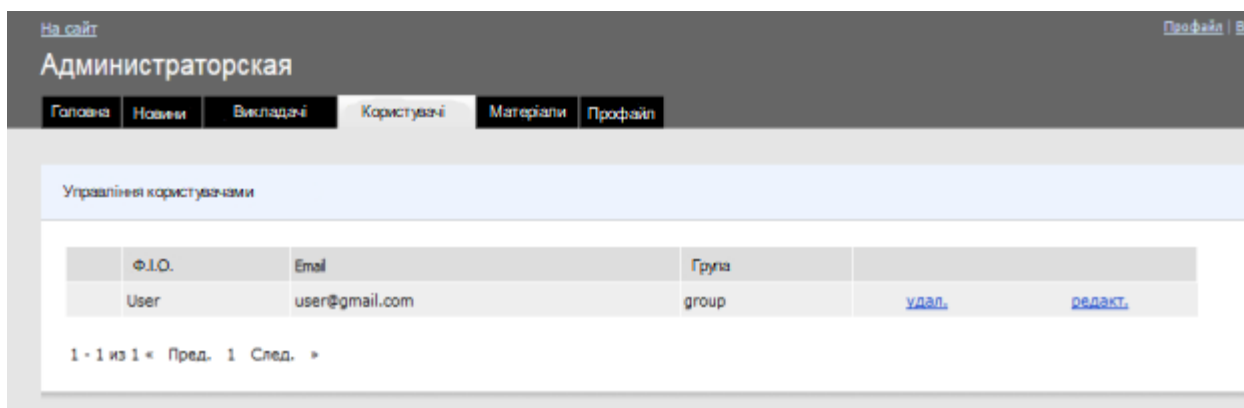


Рисунок 4.5 – Управління користувачами

При додаванні матеріалу необхідно заповнити наступні поля:

- вибрати ПІБ викладача, що додає матеріал;
- предмет;
- спеціальність;
- курс;
- група;
- назва;
- анотація;
- повний текст матеріалу;
- файли матеріалу.

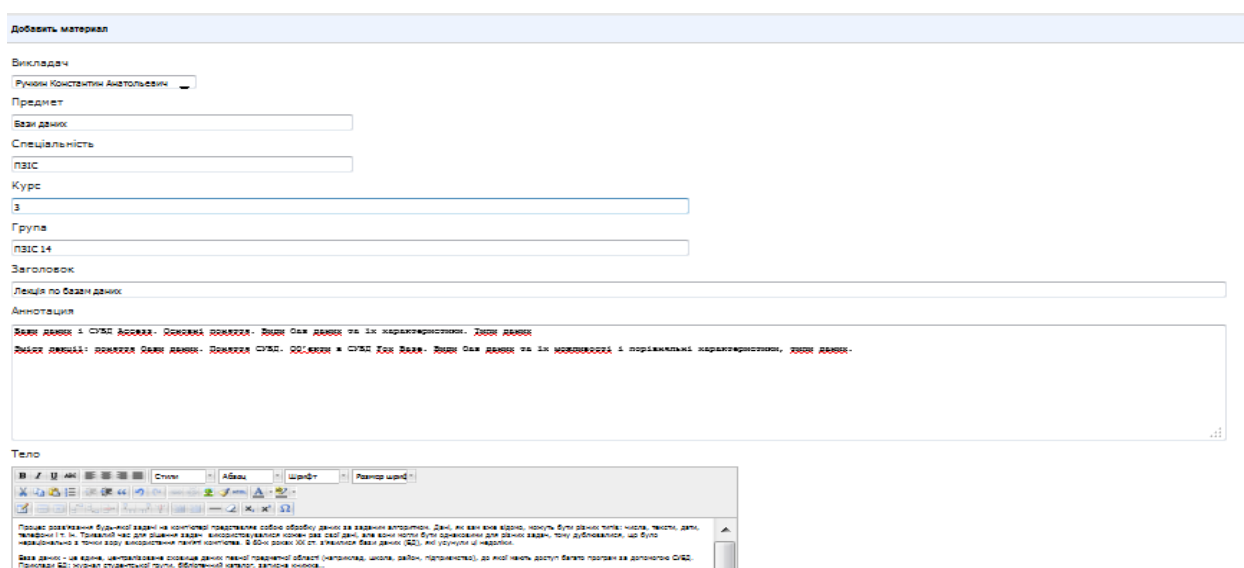


Рисунок 4.6 – Додавання матеріалів

Після додавання матеріал розміщується на головній сторінці (див. рис.4.7).

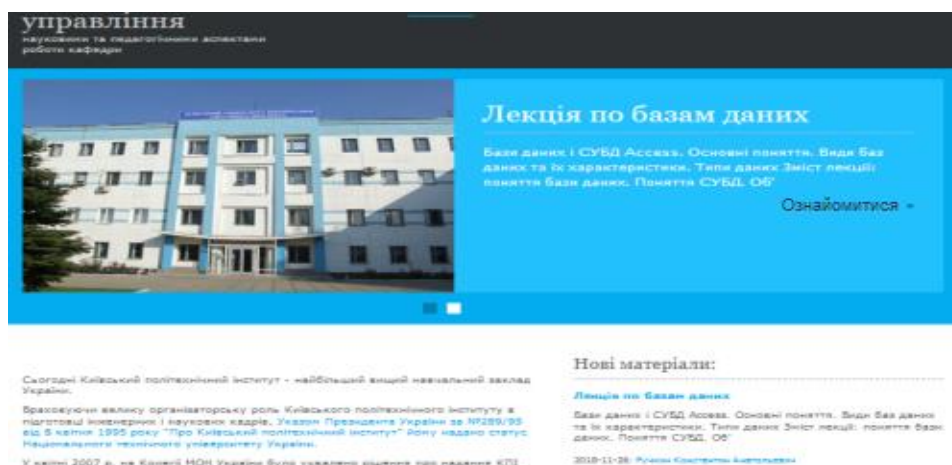


Рисунок 4.7 – Головна сторінка з доданим матеріалом

На рисунку 4.8 представлено текст завантаженого матеріалу.

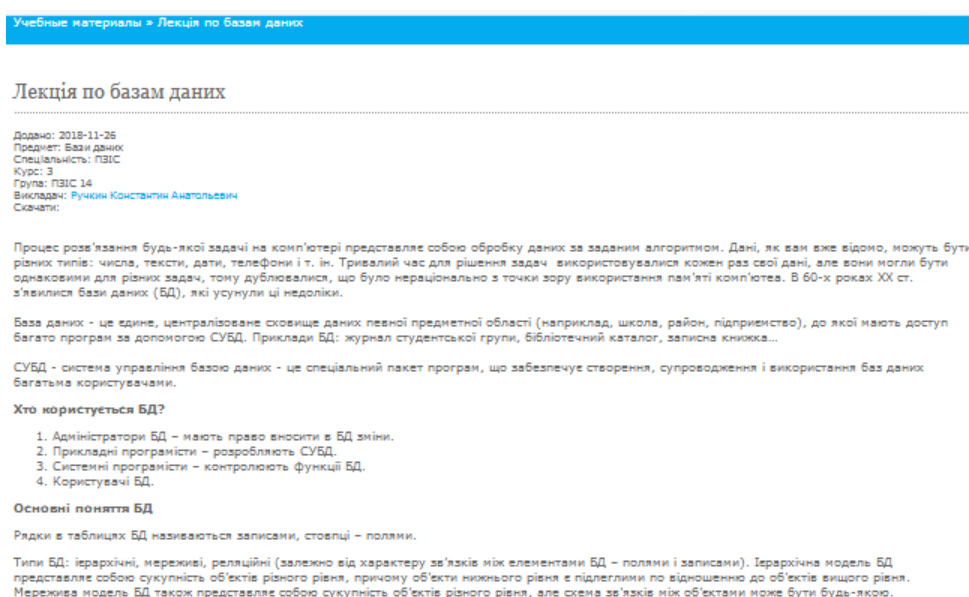
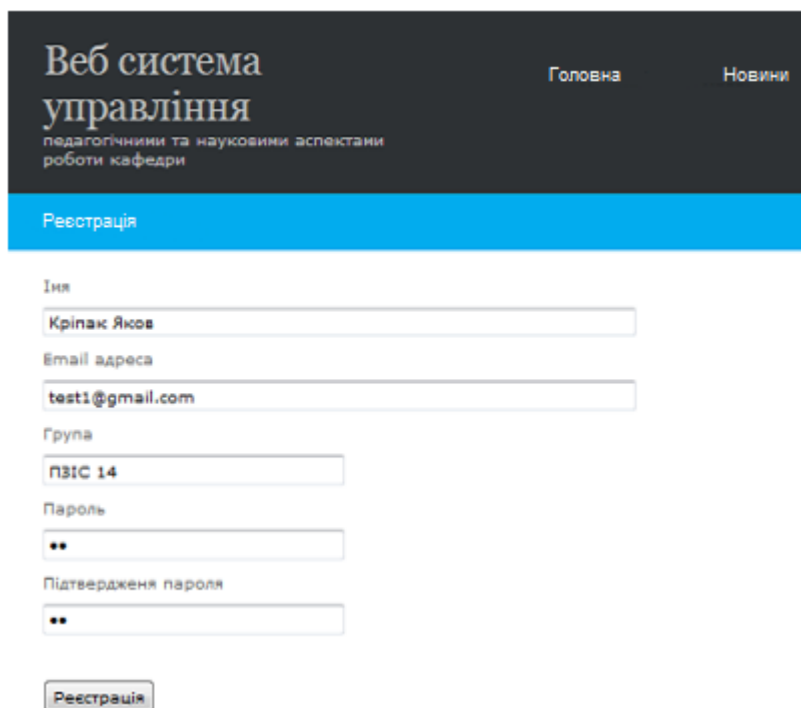


Рисунок 4.8 – Перегляд матеріалу

На рисунку 4.9 представлено форму реєстрації користувача, при реєстрації необхідно вказати:

- ім'я;
- e-mail;

- група;
- пароль;
- підтвердження паролю.



The screenshot shows a web interface for a management system. At the top, there is a dark header with the text "Веб система управління" (Web management system) and "педагогічні та наукові аспекти роботи кафедри" (pedagogical and scientific aspects of the department's work). To the right of the header are links "Головна" (Home) and "Новини" (News). Below the header is a blue bar with the word "Реєстрація" (Registration). The main form area contains several input fields: "Ім'я" (Name) with the value "Кріпак Яков", "Email адреса" (Email address) with the value "test1@gmail.com", "Група" (Group) with the value "ПЗІС 14", "Пароль" (Password) with two dots, and "Підтвердження пароля" (Confirm password) with two dots. At the bottom of the form is a button labeled "Реєстрація" (Registration).

Рисунок 4.9 – Форма реєстрації

## Висновок по розділу 4

В рамках цього розділу представлено опис функціонування розробленої веб системи управління науковими та педагогічними аспектами роботи кафедри, по наведеним прикладам можна визначити про правильність її роботи.

## 5. РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ

Ідея проекту полягає у створенні модуля повідомлення для веб системи управління педагогічними та науковими роботами кафедри. Така система допоможе полегшити процес надання необхідних для навчання та успішної наукової роботи матеріали, дасть можливість обміну повідомленнями між учасниками навчання. Ця система дозволить викладачам оптимізувати роботу зі студентами, що навчаються на кафедрі, студентам отримати необхідні для навчання дані.

### 5.1 Опис ідеї стартап-проекту

Проаналізуємо зміст ідеї, її можливі напрямки застосування, відмінність запропонована ідея від існуючих аналогів, а також основні переваги, які може отримати користувач системи [35]. Результати аналізу представлені у таблиці 5.1.

Таблиця 5.1. Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Створення модуля повідомлень веб системи управління науковими та педагогічними аспектами роботи кафедри	Представлення необхідних матеріалів для навчання та наукової роботи	Оптимізація процесу навчання
	Обмін повідомлень між учасниками процесу	Надання необхідних матеріалів для навчання та науки

На ринку існують різноманітні поштові сервіси, сховища файлів або соціальні мережі, але вони або вузькоспеціалізовані або занадто функціональні та не можуть бути застосовані при роботі кафедри. Ці аналоги, в більшості, є доступними великий кількості користувачів, що в свою чергу сприяє розповсюдженню матеріалів викладачів кафедри студентам, які не повинні мати доступ к ним.



Доцільно провести аналіз потенційних техніко-економічних переваг ідеї порівняно з пропозиціями конкурентів. Результат аналізу у таблиці 5.2.

Таблиця 5.2. Визначення характеристик ідеї проекту

Техніко-економічні характеристики ідеї	Продукція конкурентів			Слабкі (W), нейтральні (N) та сильні (S) сторони		
Назва продукту	Розроблена система	Google App Engine	Viber	W	N	S
Операційна система та версії	Крос платформена	Платформи Windows, MacOS, Android	Android			
Системні вимоги	Мінімальні	Від 2 Гб ОЗУ	Від 1 Гб ОЗУ			
Мови програмування	PHP	Python	Python			
Необхідність встановлення додаткового ПЗ	наявність АПК	наявність АПК	наявність АПК			
Ціна	безкоштовний	\$1,000	безкоштовна			

Автоматизована система вже розроблена та представлена у вигляді веб-сайту. Розроблена система є кроссплатформенною, потребує мінімальних системних вимог, а також потребує підключення до Інтернету.

Перевагами даної розробки є те, що подібні веб системи управління в світі створені для більш масштабних проектів, а ціни на такі програмні системи є

зависокими для користувачів. На вітчизняному ринку аналогів створеній системі взагалі не виявлено.

## 5.2 Технологічний аудит ідеї проекту

Для проведення технічного аудиту ідеї проекту, потрібно провести аудит технології [35], за допомогою якої можна реалізувати ідею проекту, визначити чи доступні ці технології, та чи потребують вони допрацювання. Результат представлений у таблиці 5.3.

Таблиця 5.3. Технологічна здійсненність ідеї проекту

Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
Модуль повідомлення веб системи управління науковими та педагогічними аспектами роботи кафедри	Мова сценарію PHP		Доступно (безкоштовна базова версія)
	СУБД MySQL		Доступно (безкоштовне)

Обрані технології для реалізації ідеї проекту: мова сценаріїв PHP та СУБД MySQL, які використовуються для розробки веб додатків різної ступені функціональності та призначення[49].

Обрані технології є доступними, не потребують доопрацювань, а також безкоштовні та надають усі необхідні можливості для реалізації поставленої задачі.

### 5.3 Аналіз ринкових можливостей запуску стартап-проекту

Визначення ринкових можливостей, які можна використати під час ринкового впровадження проекту, та ринкових загроз, які можуть перешкодити реалізації проекту, дозволяє спланувати напрями розвитку проекту із урахуванням стану ринкового середовища[43], потреб потенційних клієнтів та пропозицій проектів-конкурентів [35]. Проведемо аналіз попиту (таблиця 5.4).

Таблиця 5.4. Попередня характеристика потенційного ринку стартап-проекту

Показники стану ринку	Характеристика
Загальна потреба в продукції	Необхідна
Можливі річні обсяги випуску в натуральних показниках	До 100 копій
Річні обсяги випуску в вартісних показниках	10000 – 12000\$
Динаміка ринку (якісна оцінка)	Зростає
Наявність обмежень для входу	Бажання розробників працювати лише над власним ПЗ
Специфічні вимоги до стандартизації та сертифікації	Для ПЗ відсутні. Для коректної роботи - використання стандартів ISO 9126 та ISO 25010
Середня норма рентабельності в галузі (або по ринку)	79%

За попередньою оцінкою існуючий ринок здається достатньо привабливим для входження. При проведенні збору статистичних даних, що свідчать про підвищення попиту на подібні розробки, через збільшення інтересу до веб додатків як з боку педагогічних робітників, так і з боку студентів або учнів.

Враховуючи великий попит у розробці подібних програмних комплексів, для покращення процесів розробки програмних продуктів та полегшення роботи розробникам, доцільно і має сенс розробка даної веб орієнтованої системи управління науковими та педагогічними аспектами роботи кафедри.

Надалі визначаються потенційні групи клієнтів, їх характеристики, та формується орієнтовний перелік вимог до товару для кожної групи (таблиця 5.5).

Таблиця 5.5. Характеристика потенційних клієнтів стартап-проекту

Потреба, що формує ринок	Цільова аудиторія	Особливості поведінки споживачів	Вимоги споживачів до товару
Представлення наукових та навчальних матеріалів кафедри	Викладач кафедри	Викладачі потребують в використанні платформи, що дає можливість надання матеріалу учням, яким вони безпосередньо викладають	доступна ціна; зручність і простота використання; мобільність
	Студенти, що навчаються на кафедрі	Потребують в навчальних та наукових матеріалах, особливо з застосуванням віддаленого доступу	зручність і простота використання

Після визначення потенційних груп клієнтів необхідно провести аналіз ринкового середовища: скласти таблиці факторів, що сприяють ринковому впровадженню проекту, та факторів, що йому перешкоджають. Результати представлені у таблицях 5.6 та 5.7 відповідно.

Надалі необхідно провести аналіз пропозиції – визначити загальні риси конкуренції на ринку, а саме визначити тип можливої майбутньої конкуренції та її інтенсивність, рівень конкурентоспроможності за рівнем конкурентної боротьби, видами товарів і галузевою ознакою (таблиця 5.8).

Таблиця 5.6. Фактори загроз

Фактор	Зміст загрози	Можлива реакція компанії
Поява конкурентів	Можлива поява конкурентів, які спроможуться створити більш якісний чи дешевший продукт.	Постійна розробка удосконалень, розширення асортименту (додавання нових можливостей, нового функціоналу та/або додання можливостей введення нових параметрів)
Зміни тенденцій ринку	Поява більш досконалої програмної системи від конкурентів, які значно довше на ринку.	Розробка нових сучасних необхідних удосконалень, тобто додання або заміна старого функціоналу на можливості розрахунку нових параметрів
Економічний спад	Відсутність попиту на програмний продукт через економічну складову	Зменшення ціни; зміна цільової аудиторії.
Зниження репутації компанії	Можлива ситуація, коли конкуренти спроможуться на більший попит	Проведення рекламних та промо-акцій для програмного продукту

Таблиця 5.7. Фактори можливостей

Фактор	Зміст загрози	Можлива реакція компанії
Невелика кількість конкурентів	На сьогоднішній день на ринку дуже незначна кількість конкурентів, їх програмні продукти в переважній більшості більш об'ємні та складні у використанні	Розповсюджувати створений продукт, розвивати його можливості, розширювати функціонал
Відповідні тенденцій ринку	На сьогоднішній день ІТ-ринок потребує та надає всі можливості для впровадження систем, які надаватимуть користувачам можливість управління науковими та педагогічними аспектами роботи кафедри	Розповсюджувати створений продукт, розвивати його можливості, розширювати функціонал
Можливість побудови власної репутації	Новий «гравець» на ринку має всі можливості для побудови власної репутації з «чистого листка»	Пошук замовників, можливих покупців створеного програмного продукту, розширення бази замовників. Зарекомендувати себе, як надійну компанію.

Таблиця 5.8. Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	У чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії)
--------------------------------------	---	---

Таблиця 5.8 (Продовження)

Тип конкуренції	Чиста Залежить від кількості конкурентів та якості надання ними послуг у порівнянні з послугами компанії	Покращення власного продукту через зниження ціни; підвищення якості програмного продукту
За рівнем конкурентної боротьби	Локальна Конкуренція на вітчизняному ринку	На вітчизняному ринку конкурентів не виявлено, компанія має можливість встановлення власної бажаної ціни, та нарощувати клієнтську базу.
За галузевою ознакою	Внутрішньогалузева Продукт націлений лише на конкретну сферу діяльності	Немає можливостей та сенсу розширювати функціонал за межі використання в начальних закладах.
Конкуренція за видами товарів	Марки-конкуренти Створений товар може мати конкурентів, які пропонують аналогічний товар	Зниження ціни, розширення функціональних, реклама для популяризації програмного продукту
За характером конкурентних переваг	Цінова Важливо за скільки продається товар, та його прибуток	Можливе підвищення ціни на нові розробки, зниження на старі версії для заохочення покупців
За інтенсивністю	Марочна Можуть з'являтися конкуренти	При виході на міжнародний ринок потрібно рекламувати кращий функціонал продукту та доводити свою надійність

Після аналізу конкуренції проводиться більш детальний аналіз умов конкуренції в галузі (таблиця 5.9) - за моделлю п'яти сил М. Портера, який вирізняє п'ять основних факторів, що впливають на привабливість вибору ринку з огляду на характер конкуренції:

- конкурент, що вже є у галузі;
- потенційні конкуренти;
- наявність товарів-замінників;
- постачальники, що конкурують за ринкову владу;
- споживачі, які конкурують за ринкову владу.
- 

Таблиця 5.9. Аналіз конкуренції в галузі за М.Портером

Складові галузі	Прямі конкуренти в галузі	Потенційні конкуренти	Клієнти	Товари-замінники
	Розробники аналогічних систем	Кращі продукти, ширший функціонал	Мають найбільше значення. Більш важлива їх кількість, ніж постійна співпраця	Є. Відрізняються більшими вимогами до ресурсів, мають значно більший функціонал, який не застосовується у цій області



Таблиця 5.9 (Продовження)

Висновки	Інтенсивність конкурентної боротьби з боку прямих конкурентів незначна	Наявні усі можливості входу на ринок. Потенційні конкуренти не виявлені. Строки виходу на ринок – один день	Необхідність клієнтської-бази, тому важливо знаходити можливості приваблення споживачів до власного продукту	Немає обмежень

На основі аналізу конкуренції за М. Портером, проведеного у таблиці 5.9, а також із враховуючи характеристики ідеї проекту (таблиця 5.2), вимог споживачів до товару (таблиця 5.5) та факторів маркетингового середовища (таблиці 5.6 і 5.7) визначимо та обґрунтуємо перелік факторів конкурентоспроможності (таблиця 5.10).

Таблиця 5.10. Обґрунтування факторів конкурентоспроможності

Фактор конкурентоспроможності	Обґрунтування
Невелика кількість конкурентів на ринку	На вітчизняному ринку (на який для початку націлена розроблена програмна система), конкурентів немає
Доступність створеного продукту (програмно)	Немає жорстких системних вимог, програма буде працювати на всіх пристроях, що підтримують Інтернет.
Легкість і простота використання	Зручний зрозумілий інтерфейс, створені довідка та інструкція для користувача
Підключення до мережі Інтернет	Є потреби у підключенні до мережі Інтернет після придбання продукту, як і в більшості аналогів

Таблиця 5.10 (Продовження)

Потреба у постійному супроводі	Не потребує додаткового супроводу спеціалістів
Додаткові компоненти	Немає необхідності встановлення додаткових компонентів, на відміну від деяких аналогів

За визначеними факторами конкурентоспроможності проводиться аналіз сильних та слабких сторін стартап-проекту, проведений у таблиці 5.11.

Таблиця 5.11. Порівняльний аналіз сильних та слабких сторін проекту

Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів						
		-3	-2	-1	0	+1	+2	+3
Мала кількість / відсутність конкурентів	10				+			
Системні вимоги	20			+				
Простота використання	19	+						
Не потрібен супровід	11					+		

Фінальним етапом ринкового аналізу можливостей впровадження проекту є складання SWOT-аналізу (матриці аналізу сильних (Strength) та слабких (Weak) сторін, загроз (Troubles) та можливостей (Opportunities) на основі виділених ринкових загроз та можливостей, та сильних і слабких сторін (таблиця 5.12).

Таблиця 5.12. SWOT-аналіз проекту

Сильні сторони (S):	Слабкі сторони (W):
невелика ціна; інноваційні технології молодий і перспективний колектив; гнучка політика керівництва;	брак власного устаткування; брак робочої сили; недостатньо оборотних коштів; відсутність репутації компанії;

Таблиця 5.12 (Продовження)

Можливості (О):	Загрози (Т):
додаткові послуги; вихід на нові ринки; розширення клієнтської бази; співпраця з іншими компаніями.	поява нових конкурентів; зниження репутації компанії; економічний спад; зміни тенденцій попиту.

На основі SWOT-аналізу розробимо альтернативу ринкової поведінки для виведення стартап-проекту на ринок та орієнтовний оптимальний час їх ринкової реалізації з огляду на потенційні проекти конкурентів, що можуть бути виведені на ринок.

Визначені альтернативи аналізуються з точки зору строків та ймовірності отримання ресурсів (таблиця 5.13).

Таблиця 5.13. Альтернативи ринкового впровадження стартап-проекту

Альтернатива ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
Вихід на нові ринки	Пошук інвесторів	2-4 місяців
Розширення виробничої лінії	Пошук інвесторів	Після виходу на ринок основного продукту, до 5 місяців

Отже, після проведеного аналізу вирішено спочатку вивести на основний ринок розроблену систему, а вже потім шукати можливості розширення програмного функціоналу для користувачів.

## 5.4 Розроблення ринкової стратегії проекту

Розробка ринкової стратегії перш за все передбачає визначення стратегії охоплення ринку [35], включаючи опис цільових груп потенційних споживачів, які визначені у таблиці 5.14.

Таблиця 5.14. Вибір цільових груп потенційних споживачів

Опис цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в сегменті	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
Викладачі кафедр, педагогічні робітники	Потребують	Попит є	Незначна	Просто
Студенти, учні	Потребують	Попит є	Незначна	Помірно

Оскільки різниця між цільовими групами зовсім незначна, а також враховуючи той факт, що компанія має бажання почати продажі (а відповідно і отримання прибутку) як найшвидше, то доцільно враховувати обидві цільові групи, тобто використовувати масовий маркетинг, пропонуючи стандартизовану програму.

За результатами аналізу потенційних груп споживачів (сегментів) автори ідеї обирають цільові групи, для яких вони пропонуватимуть свій товар, та визначають стратегію охоплення ринку.

Для роботи в обраних сегментах ринку необхідно сформувати базову стратегію розвитку, яка визначається у таблиці 5.15.

Таблиця 5.15. Визначення базової стратегії розвитку

Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
Вихід на нові ринки	Стратегія диференціації	Надання програмному продукту відмінних якостей, які роблять систему особливою на фоні аналогічних розробок	Стратегія диференціації
Розширення виробничої лінії	Стратегія диференціації (допускається стратегія спеціалізації)	Надання товару кращих властивостей та розширення функціоналу	Стратегія диференціації (допускається стратегія спеціалізації)

Вибір стратегії конкурентної поведінки визначається у таблиці 5.16.

Таблиця 5.16. Визначення базової конкурентної поведінки

Чи є проект «першопроходцем» на ринку	Ні
Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Так
Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Так, розширення функціональності, але зменшення вимог до ресурсів
Стратегія конкурентної поведінки	Стратегія виклику лідера

На основі вимог споживачів з обраних сегментів до постачальника (стартап-компанії) та до програмного продукту, а також в залежності від обраної базової стратегії розвитку та стратегії конкурентної поведінки необхідно розробити стратегію позиціонування (таблиця 5.17), що полягає у формуванні ринкової позиції, за яким споживачі мають ідентифікувати торгівельну марку або проект.

Таблиця 5.17. Визначення стратегії позиціонування

Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту
Доступна ціна, простота і зручність використання	Стратегія диференціації	Легкість і простота у використанні. Доступність через ціну та технічні характеристики. Вирішення важливих поставлених задач швидко, легко та зрозуміло навіть без інструкцій.	стандарту якості; метрики програмного забезпечення.

Отже, робота стартап-компанії на ринку повинна бути спланована орієнтовано таким чином: за стратегією диференціації виконаний і буде поширюватись відмінний програмний продукт, дотримуючись у конкурентній поведінці стратегії «виклику лідера», тобто випускається один товар для усіх можливих споживачів.

## 5.5 Розроблення маркетингової програми стартап-проекту

Першим кроком під час розробки маркетингової програми стартап-проекту є формування маркетингової концепції товару [35]. Для цього у таблиці 5.18 підсумовані результати попереднього аналізу конкурентоспроможності товару.

Таблиця 5.18. Визначення ключових переваг концепції потенційного товару

Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами
Оцінка якості ПП	Оцінка за метриками. Удосконалення оцінки будь-якої з обраних характеристик.	Розрахункові показники, точність та достовірність яких можна оцінювати; самостійність програмної системи.

Надалі розроблена трирівнева маркетингова модель товару: уточнюються ідея продукту, його фізичні складові та особливості процесу його надання (таблиця 5.19).

Таблиця 5.19. Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові
Товар за задумом	Модуль повідомлень для веб системи управління науковими та педагогічними аспектами роботи кафедри. Можливість розміщення наукового та навчального матеріалу, ідентифікація користувачів, розподілення до ресурсів. Формування повідомлень на поштову скриньку учасників навчання.
Товар у реальному виконанні	Реалізовано веб систему управління науковими та педагогічними аспектами роботи кафедри, що дозволяє розміщувати навчально-науковий матеріал викладачам, студентам користуватися ним

Таблиця 5.19 (Продовження)

Товар із підкріпленням	До продажу: стандартна розроблена система (Модуль повідомлень для веб системи управління науковими та педагогічними аспектами роботи кафедри)
	Після продажу: додані додаткові можливості

Подібна система не реалізовувалось раніше, а тому є необхідність у фіксуванні авторських прав або отриманні патенту.

Визначення цінових меж, якими необхідно керуватись при встановленні ціни на потенційний товар, яке передбачає аналіз ціни на товари-аналоги, а також аналіз рівня доходів цільової групи споживачів описано в таблиці 5.20.

Таблиця 5.20. Визначення меж встановлення ціни

Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни
1050 – 3550 \$	300 – 1500 \$	10 – 15 \$

Наступним кроком є визначення оптимальної системи збуту, в межах якого приймається рішення (таблиця 5.21): чи потрібно проводити збут власними силами або залучати сторонніх посередників, вибір та обґрунтування оптимальної глибини каналу збуту, вибір та обґрунтування виду посередників.

Останньою складовою маркетингової програми є розроблення концепції маркетингових комунікацій, що спирається на попередньо обрану основу для позиціонування, визначену специфіку поведінки клієнтів (таблиця 5.22).



Таблиця 5.21. Формування системи збуту

Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
Бажання отримати більше за менші гроші	Пошук клієнтської бази та продаж	Нульовий рівень: тільки виробник	Вертикальна маркетингова система

Таблиця 5.22. Концепція маркетингових комунікацій

Поведінка цільових клієнтів	Канали комунікацій цільових клієнтів	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення
Бажання отримати більше за менші гроші	Будь-які	Низька ціна Легкий і простий у використанні продукт	Донести до користувача суть продукту, його якість, та залучити якомога більше зацікавлених клієнтів

## Висновки до розділу 5

1. Маркетингова програма орієнтовано має бути побудована таким чином:

- створення програмного продукту;
- пошук потенційних клієнтів (користувачі, компанії);

— базова стратегія розвитку – стратегія диференціації, тобто конкурентоспроможність формується шляхом надання споживачеві бажаного товару. На основі ретельного вивчення споживчого середовища розробляється одна або декілька відмітних характеристик власного товару;

— стратегія конкурентної поведінки – стратегія виклику лідера, тобто на споживчому ринку орієнтуватись на всіх можливих споживачів, у тому числі

клієнтів фірм-конкурентів. За подальші цілі ставиться можливість обігнати лідерів цільового сегменту.

2. Конкурентні переваги створеного продукту очевидні, так як на вітчизняному ринку аналогів (а відповідно і конкурентів) не виявлено. У той час як попит на програмні системи подібного роду тільки набирає популярність. Схожі програмні розробки відрізняються або вузькою спеціалізацією – надають можливість або тільки розміщення матеріалів або тільки формування повідомлень, або відрізняються високими функціональними та ресурсними вимогами – соціальні мережі, хмарні технології.

3. Перспективи впровадження з огляду на потенційні групи користувачів, стан конкуренції та конкурентоспроможності проекту – прямі, і тільки доводять можливість впровадження, та не марну розробку створеного програмного продукту.

## ВИСНОВКИ

В рамках виконання цієї роботи були вирішені наступні задачі:

1. Проведено аналіз діяльності кафедри при виконанні наукових та педагогічних робіт кафедри, за допомогою стандартів IDEF0, DFD;
2. Визначені вимоги до побудови веб системи управління науковими та педагогічними аспектами роботи кафедри, сформовано постановку завдання;
3. Визначені основні етапи розробки веб системи, що значно полегшить виконання завдання;
4. Визначено користувачів веб системи управління науковою та педагогічною роботою кафедри, побудовані діаграми варіантів використання;
5. Побудовано алгоритмічне забезпечення задачі, представлена робота основних підсистем;
6. Проведено аналіз існуючих платформ на яких відбувається побудова веб систем управління;
7. Проведено вибір середі розробки за допомогою методу аналізу ієрархій, на основі якого визначено, що найбільш оптимальним рішенням для розробки веб системи управління науковими та педагогічними аспектами роботи кафедри є розробка за допомогою написання власного коду.
8. Виконано побудову діаграми класів;
9. Розроблено діаграму компонентів та розгортання;
10. Побудовано логічну схему БД веб системи;
11. Визначене середовище розробки веб системи управління науковою та педагогічною роботою кафедри;
12. Виконана програмна реалізація веб системи управління науковими та педагогічними аспектами роботи кафедри;
13. Побудовано карту веб системи;
14. Представлено опис функціонування розробленої веб системи.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Брендон Д. Буквар по PHP і MySQL: навч. посібник / Д. Брендон. - СПб: Пітер, 2014. - 160 с.
2. Буч, Грейді Мова UML. Керівництво користувача / Грейді Буч, Джеймс Рамбо, Айвар Джекобсон. - М.: ДМК, 2015. - 432 с.
3. Гома, Хассан UML. Проектування систем реального часу, паралельних і розподілених додатків / Хассан Гома. - М.: ДМК Пресс, 2016. - 700 с.
4. Глушаков С.В. Робота в мережі Internet С.В.. Глушаков, Д.В. Ломотько, А.С. Сурядні. - Харків: Фоліо, 2013. - 157 с.
5. Глушаков, С.В. Бази даних / С.В. Глушаков, Д.В. Ломотько. - М.: Харків: Фоліо, 2017. - 504 с.
6. Голіцина Бази даних / Голіцина, О.Л. і. - М.: Форум; Инфра-М, 2015. - 399 с.
7. Грекул, В. І. Управління впровадженням інформаційних систем / В.І. Грекул, Г.Н. Денищенко, Н.Л. Коровкина. - Москва: РДГУ, 2014. - 224 с.
8. Гришин В. О. Інтернет і бізнес. Як використовувати сучасні технології і не залишитися в дурнях ?; Книга по Требованию - Москва, 2012. - 186 с.
9. Гультьєв, А. К. Проектування і дизайн користувацького інтерфейсу / А.К. Гультьєв, В.А. Машин. - М.: Корона-Принт, 2015. - 350 с.
10. Дрігалкін В.В. HTML в прикладах. Як створити свій Web-сайт: самоучитель В.В.. Дрігалкін. - М.: Діалектика, 2013. - 167 с.
11. Карпова, Т.С. Бази даних. Моделі, розробка, реалізація / Т.С. Карпова. - М.: СПб: Пітер, 2016. - 304 с.
12. Ларман, Крег Застосування UML 2.0 і шаблонів проектування. Введення в об'єктно-орієнтований аналіз, проектування і ітеративну розробку / Крег Ларман. - М.: Вільямс, 2013. - 736 с.
13. Малихіна, М. Бази даних: основи, проектування, використання / М. Малихіна. - М.: БХВ-Петербург, 2016. - 512 с.

14. Мюллер, Роберт Дж. Бази даних та UML. Проектування / Р.Дж. Мюллер. - М.: ЛОРИ, 2017. - 420 с.
15. Мюллер, Роберт Дж. Проектування баз даних та UML / Р.Дж. Мюллер. - М.: ЛОРИ, 2013. - 422 с.
16. Дронов В. PHP, MySQL і Dreamweaver. Розробка інтерактивних Web-сайтів; БХВ-Петербург - М., 2017. - 480 с.
17. Жадаєв Олександр PHP для початківців; "Видавництво" Пітер "- М., 2014. - 288 с.
18. Качанов А.Н. Буквар по PHP і MySQL: навч. посібник О.М.. Качанов. - М.: Інфо, 2010. - 129 с.
19. Редько, В. Н. Бази даних та інформаційні системи / В.М. Редько, І.А. Басараб. - М.: Знание, 2017. - 341 с.
20. Савельєва Н.В. Основи програмування на PHP /Н.В. Савельєва. - Томськ: ІНТУЇТ, 2005. - 289 с.
21. Ховард Майкл, Лебланк Девід, Виега Джон Як написати безпечний код на C ++, Java, Perl, PHP, ASP.NET; ДМК Прес - М., 2014. - 288 с.
22. Храмцов П.Б. Основи web-технологій /П.Б. Храмцов. - Томськ: ІНТУЇТ, 2003. - 295 с.
23. Шаймарданов, Р.Б. Моделювання та автоматизація проектування структур баз даних / Р.Б. Шаймарданов. - М.: Радио и связь, 2017. - 120 с
24. Шкриль А. PHP - це просто. Програмуємо для Web-сайту; БХВ-Петербург - М., 2006. - 368 с.
25. Яргер, Р.Дж. MySQL і mSQL: Бази даних для невеликих підприємств і Інтернету / Р.Дж. Яргер, Дж. Різ, Т. Кінг. - М.: СПб: Символ-Плюс, 2014. - 560 с.
26. Колісниченко, Д.Н. Вибираємо кращий безкоштовний движок для сайту. CMS Joomla! і Drupal (+ CD-ROM) / Д.М. Колісниченко. - М.: БХВ-Петербург, 2016. - 184 с.
27. Seamus, Bellamy Joomla! For Dummies® / Seamus Bellamy. - Москва: СПб. [и др.] : Питер, 2015. - 360 с.

28. Норт, Беррі Joomla! 1.5. Практичний посібник / Беррі Норт. - М .: Символ-плюс, 2017. - 464 с.
29. Вічугова, А. А. Методи і засоби концептуального проектування інформаційних систем: порівняльний аналіз структурного і об'єктно-орієнтованого підходів / А.А. Вічугова. - М .: Синергія, 2014. - 631 с.
30. Іпатова, Е. Р. Методології і технології системного проектування інформаційних систем. Підручник: моногр. / Е.Р. Іпатова. - М .: Флінта, 2016. - 300 с
31. Мартишин, Сергій Анатолійович Проектування і реалізація баз даних в СУБД MySQL з використанням MySQL Workbench. Навчальний посібник. Методи і засоби проектування інформаційних систем і технологій. Інструментальні засоби інформаційних мереж. Гриф УМО вузів Росії / Мартишин Сергій Анатолійович. - М .: Форум, 2017. - 570 с.
32. Калашян, А.Н. Структурные модели бизнеса: DFD-технологии / А.Н. Калашян. - М.: Финансы и статистика, 2009. - 599 с
33. Michael Morrison, Lynn Beighley Head First PHP & MySQL / O'Reilly Media, Inc., 2008. - 125 с.
34. Tom Butler, Kevin Yank PHP & MySQL: Novice to Ninja, 6th Edition / SitePoint Pty. Ltd., 2017. - 432 с.
35. Luke Welling, Laura Thompson PHP & MySQL Web Development / Sams Ltd., 2016. – С. 245-247с.
36. Robin Nixon Learning PHP, MySQL & JavaScript / O'Reilly Media, Inc., 2014. - 166 с.
37. Vikram Vaswani PHP A Beginner's Guide / The McGraw-Hill Companies, 2009 - 232 с.
38. Brett McLaughlin PHP & MySQL / O'Reilly Media, Inc., 2012 – С/ 123-127.
39. Joel Murach, Ray Harris Murach's MySQL, 2nd Edition / Mike Murach & Associates, 2014 – С. 312-325.
40. Paul DuBois MySQL Cookbook / O'Reilly Media, Inc., 2014 – С. 784-795.
41. Baron Schwartz, Peter Zaitsev, Vadim Tkachenko High Performance MySQL / O'Reilly Media, Inc., 2012 – С. 635-639.

42. Andrew Comeau, Stephen Burge MySQL Explained: Your Step By Step Guide / CreateSpace Independent Publishing Platform, 2015 – C. 222-224.
43. George Reese MySQL Pocket Reference / O'Reilly Media, Inc., 2007 – 76 c.
44. Sveta Smirnova MySQL Troubleshooting: What To Do When Queries Don't Work / O'Reilly Media, Inc., 2012 C. 211-212
45. Antun Peicevic MySQL introduction / CreateSpace Independent Publishing Platform, 2016 C. 32-33.
46. Sasha Pachev Understanding MySQL Internals: Discovering and Improving a Great Database / O'Reilly Media, Inc., 2007 243 c.
47. Guy Harrison MySQL Stored Procedure Programming: Building High-Performance Web Applications in MySQL / O'Reilly Media, Inc., 2006 631 c.
48. Seyed M.M., Tahaghoghi, Hugh Williams Learning MySQL: Get a Handle on Your Data / O'Reilly Media, Inc., 2006 515 c.
49. Charles Bell, Mats Kindahl, Lars Thalmann MySQL High Availability: Tools for Building Robust Data Centers / O'Reilly Media, Inc., 2014 712 c.
50. Baron Schwartz, Peter Zaitsev, Vadim Tkachenko High Performance MySQL: Optimization, Backups, and Replicatio / O'Reilly Media, Inc., 2012 801 c.

# ДОДАТОК А

Апробації

Консолідація даних з використанням стохастичних методів оптимізації

УКР.НТУУ”КПІ”\_ТЕФ\_АПЕПС\_ ТР31239\_18М

Аркушів 4

2018





УДК 524.36

**Сучасні проблеми наукового забезпечення енергетики:** Матеріали XVI Міжнародної науково-практичної конференції аспірантів, магістрантів і студентів, м. Київ, 24–27 квітня 2018 р. У 2 т. – К. : 7 КПІ ім. Ігоря Сікорського, 2018. – Т. 2. – 298 с.

**ISBN 978-966-622-886-7**

**ISBN 978-966-622-888-1 (Т.2)**

Подано тези доповідей XVI Міжнародної науково-практичної конференції аспірантів, магістрантів і студентів «Сучасні проблеми наукового забезпечення енергетики» за напрямками: автоматизація теплоенергетичних процесів, геометричне моделювання та проблеми візуалізації, програмне забезпечення інформаційних систем та мережних комплексів, моделювання та аналіз теплоенергетичних процесів.

**Головний редактор**

Є.М. Письменний, д-р техн. наук, проф.

**Заступник головного редактора**

Ю.Є. Ніколаєнко, д-р техн. наук, с.н.с.

**Редакційна колегія:**

О.Ю. Черноусенко, д-р техн. наук, проф.,

Г.Б. Варламов, д-р техн. наук, проф.,

О.В. Коваль, канд. техн. наук, доц.,

В.О. Туз, д-р техн. наук, проф.,

О.В. Степанець, канд. техн. наук, доц.,

П.О. Барабаш, канд. техн. наук, доц.,

П.П. Меренгер, ст. викладач,

Р.П. Саков, асистент,

С.Г. Карпенко, канд. фіз.-мат. наук, доц.,

І.А. Остапенко, асистент,

М.В. Воробйов, канд. техн. наук, асистент,

О.С. Алексеїк, асистент.

**Відповідальний секретар**

О.В. Авдєєва.

*Друкується в авторській редакції за рішенням Вченої ради  
теплоенергетичного факультету Національного технічного університету  
України «Київський політехнічний інститут імені Ігоря Сікорського»  
(протокол № 8 від 26 березня 2018 р.)*

© Автори тез доповідей, 2018

**ISBN 978-966-622-886-7**

**ISBN 978-966-622-888-1 (Т.2)**

- Дубровська В.В., 229, 233  
 Дудкін Ю.М., 136  
 Дудник В.Ю., 224  
 Дудник С.О., 47, 48  
 Дядюра В.В., 22, 25  
 Євтушенко А.М., 216  
 Єдинак Ю.І., 171  
 Єлісєєв А.Г., 124  
 Єщенко О.І., 236, 253  
 Жиров М.І., 270  
 Жорновий Е.Г., 254  
 Завертана І.Я., 83  
 Завістовська А.І., 156  
 Залевський Б.І., 59  
 Зарицький В.П., 157  
 Заставнюк Д.О., 60, 61  
 Захарченко А.С., 63  
 Защик С.М., 172  
 Зіліцький В.С., 175  
 Іванов М.О., 266  
 Іванюк А.О., 217  
 Ільчишин Д.В., 248  
 Канівець О.В., 176  
 Караєва Н.В., 244, 245, 255  
 Карпенко Д.І., 137  
 Карпенко Є.Ю., 184, 185, 189, 190, 194, 204  
 Карпенко С.Г., 121, 122, 130, 148, 171, 172, 173  
 Касьяненко І.І., 190  
 Каценко Т.С., 72  
 Кириченко О.О., 158  
 Климець А.С., 159  
 Коваль О.В., 239, 240, 241, 256  
 Ковальчук А.М., 263, 265  
 Ковальчук Г.О., 49  
 Ковальчук Д.О., 29  
 Ковецький М.М., 102  
 Кокотова Д.О., 138  
 Коломоець І.М., 220  
 Колот С.С., 191  
 Колумбет В.П., 89, 108, 110, 113  
 Кондратенко Д.А., 23  
 Кондратенко І.І., 255  
 Кондрашов К.В., 256  
 Кондренко В.А., 17  
 Конкіна Н.С., 103  
 Корнійчук М.А., 73  
 Костенко І.П., 74  
 Костенко О.П., 104  
 Костюк Є.І., 257  
 Кошмак Є.С., 26  
 Крайнев І.В., 192  
 Крамар Ю.М., 208  
 Крепак О.В., 139  
 Кривда О.В., 68  
 Крижанівська Ю.В., 193  
 Крижановський К.С., 24, 39  
 Криштапович І.О., 160  
 Крючковська А.В., 105  
 Крячок О.С., 77  
 Кублій Л.І., 74, 78, 87, 93, 104, 114, 115  
 Кузьменко І.М., 116  
 Кузьмініх В.О., 226, 234, 235, 246, 262, 264  
 Кулешов М.М., 125  
 Курсенко Л.О., 258  
 Кучер В. А., 259  
 Лабжинський В.А., 192, 197, 205  
 Лалак Б.О., 126  
 Левченко Л.О., 79, 95  
 Лейбіченко О.Ю., 207  
 Лєконцева О.Е., 225  
 Лисенко Д.В., 84  
 Листопадов Д.О., 239  
 Лисяний Є.С., 140  
 Литвиненко Д.С., 141  
 Логвін М.А., 85  
 Лукомський Я.Ю., 5  
 Лучков В.Ф., 194  
 Любицький С.В., 12, 21, 23, 30, 31, 60  
 Ляшенко М.В., 207, 213  
 Магерамова І.І., 240  
 Мажара О.О., 206  
 Малишев М.С., 195  
 Малюх О.А., 161  
 Марич Т.І., 162  
 Маріяш Ю.І., 13  
 Мартікян А.К., 163  
 Маруня А.В., 164  
 Марчишина О.В., 241  
 Масечко І.О., 86  
 Медведєва В.М., 178, 179, 200  
 Мельник О.В., 75  
 Мельниченко А.В., 208  
 Миرونчук А.В., 37  
 Михайлова І.Ю., 81, 86  
 Михалько В.Г., 221  
 Мікульська М.А., 173  
 Мінтус М.А., 177  
 Міхеєв О.С., 260  
 Молодід О.К., 218, 219, 220, 221  
 Морозов Д.С., 142  
 Морозов М.С., 242

**УДК 004.4**

Магістрант 5 курсу, гр. ТР-71мп Лучков В.Ф.  
Доц., к.т.н. Карпенко С.Ю.

**МОДУЛЬ ВНУТРІШНІХ ПОВІДОМЛЕНЬ ВЕБ - СИСТЕМИ УПРАВЛІННЯ  
ПЕДАГОГІЧНИМИ ТА НАУКОВИМИ АСПЕКТАМИ РОБОТИ КАФЕДРИ**

На сьогоднішній день можна спостерігати проблему комунікації між студентами та викладачами під час вибору дипломного керівника та взагалом, тому що відсутня можливість віддалено домовлятися про щось із викладачами. Не існує єдиного способу віддалено проводити комунікацію із різними викладачами.

Вирішенням цієї проблеми може бути централізована веб система, що забезпечуватиме комунікацію між викладачами та студентами, яка потенційно може бути впроваджена на сайт кафедри.

Система дозволить відправляти відформатовані повідомлення одному або декільком користувачам, з можливістю вказати тему, прикріпити файли або зображення. Також буде реалізовано можливість навігації по історії листувань.

Для реалізації інтерфейсу даної системи було вирішено використовувати AngularJS 1.6 [1]. Це дозволить створити зручний та сучасний веб інтерфейс, та допоможе використовувати шаблон MVC на клієнті. Для побудови дизайну інтерфейсу вирішено використовувати підхід Material Design [2] розроблений компанією Google, що є системою об'єднання стилю, брендингу, взаємодії та руху компонентів за послідовним набором принципів.

Усі сучасні складні клієнти написані на мові JavaScript необхідно збирати задля оптимізації коду та зменшення розміру і кількості файлів, які браузер буде завантажувати, коли користувач заходить на сторінку, що в результаті призведе до пришвидшення завантаження сторінки. Для цього одним з найкращих рішень є інструмент збірки Browserify [3] та менеджер задач Gulp [4]. Для того, щоб, досить легко домогтися однаково правильної роботи застосунку у різних браузерах, таких як Internet Explorer, Microsoft Edge, Chrome, Firefox, Safari, необхідно буде використовувати програму Babel, та поліфіли.

Babel перепише JavaScript код із стандарту ES6 у код передостаннього стандарту ES5, який підтримують старі браузери.

Browserify реалізує підтримку модульної збірки SystemJS, а також автоматично вирішує залежності між модулями програми. Таким чином Browserify допоможе вирішити проблему розподілення та повторного використання коду.

Таким чином обґрунтовано необхідність та потенційну користь веб системи для комунікації, що може значно заощадити час та полегшити спілкування між викладачами та студентами.

Перелік посилань:

1. AngularJS [Електронний ресурс] — Режим доступу: <https://angularjs.org/>  
<https://www.statista.com/topics/2430/smart-homes/>
2. Material Design [Електронний ресурс] — Режим доступу: <https://material.io/>
3. Browserify [Електронний ресурс] — Режим доступу: <http://browserify.org/>
4. Gulp [Електронний ресурс] — Режим доступу: <https://gulpjs.com/>

## **ДОДАТОК Б**

Акт впровадження

Консолідація даних з використанням стохастичних методів оптимізації

УКР.НТУУ"КПІ"\_ТЕФ\_АПЕПС\_ ТР31239\_18М

Аркушів 2

2018